

셀룰러 토폴로지를 이용한 프로그레시브 솔리드 모델 생성 및 전송

이재열*, 이주행**, 김 현**, 김형선**

Generation and Transmission of Progressive Solid Models Using Cellular Topology

Lee, J. Y.*, Lee, J.-H.**, Kim, H.** and Kim, H.-S.**

ABSTRACT

Progressive mesh representation and generation have become one of the most important issues in network-based computer graphics. However, current researches are mostly focused on triangular mesh models. On the other hand, solid models are widely used in industry and are applied to advanced applications such as product design and virtual assembly. Moreover, as the demand to share and transmit these solid models over the network is emerging, the generation and the transmission of progressive solid models depending on specific engineering needs and purpose are essential. In this paper, we present a Cellular Topology-based approach to generating and transmitting progressive solid models from a feature-based solid model for internet-based design and collaboration. The proposed approach introduces a new scheme for storing and transmitting solid models over the network. The Cellular Topology (CT) approach makes it possible to effectively generate progressive solid models and to efficiently transmit the models over the network with compact model size. Thus, an arbitrary solid model SM designed by a set of design features is stored as a much coarser solid model SM^0 together with a sequence of n detail records that indicate how to incrementally refine SM^0 exactly back into the original solid model $SM = SM^n$.

Key words : Progressive solid model, cellular topology, feature-based design, level-of-details (LOD), solid model

1. 서 론

컴퓨터 그래픽스 분야에서 실감에 대한 기대가 높을수록 이를 만족시키기 위한 형상 모델의 섬세함이 요구된다. 모델링 시스템에서 섬세한 모델은 Extrusion, Constructive Solid Geometry, Freeform Deformation 등과 같은 다양한 형상 오퍼레이션에 의해서 생성될 수 있다. 하지만 효율적인 가시화를 위해서는 이렇게 생성된 모델이 삼각망과 같은 메쉬로 격자화(Tessellation)되어야만 한다. 그러나, 이렇게 생성된 메쉬는 저장, 전송, 렌더링(Rendering) 등을 지원하기에는 많은 어려움이 따르며, 이를 극복하기 위해서 1) 메쉬 단순화(Mesh Simplification), 2) LOD 생성(Level-of-Detail Generation), 3) 프로그레시브 전송

(Progressive Transmission), 및 4) 메쉬 압축(Mesh Compression) 등에 관한 연구가 필요하다^[5,7,14,15].

컴퓨터 그래픽스에서 사용되는 모델은 주로 삼각망을 이용해서 표현된다. 하지만, 산업현장에서 제품개발에 (즉, 제품설계 및 가상 어셈블리) 사용되는 것은 삼각메쉬가 아니라 솔리드 모델이다. 뿐만 아니라, 이러한 제품개발 프로세스가 분산환경하에서 진행되기 때문에 솔리드 모델을 네트워크 상에서 어떻게 효율적으로 공유 및 교환을 할 것인가가 중요한 이슈로 떠오르고 있다^[4,13,21]. 이를 위해서는 특징 어플리케이션에 적합한 LOD생성 및 효율적인 전송과정이 요구된다. 하지만, 현재의 대부분의 연구결과는 삼각메쉬에 국한된 것이고, 솔리드 모델과 관련된 연구는 미미하다. 결국, 인터넷 환경하에서 제품 개발을 위한 협업이 올바르게 이루어지기 위해서는 솔리드 모델의 LOD 생성 및 프로그레시브 전송이 필수적이다^[11-13].

엔지니어는 파트를 설계할 때 이를 기능 분석을 통하여 중요한 엔티티를 설정하고 이를 설계형태로 표

*전남대학교 산업공학과
**한국전자통신연구원 분산협업기술연구팀
- 논문투고일: 2003. 03. 22
- 심사완료일: 2003. 06. 23

현한다. 따라서, 설계된 파트의 상세설계 모델은 가시화, 분석, 협업 등의 목적에 따라 새롭게 표현 및 변환되어야 할 필요가 있다. 예를 들면, 특정 어플리케이션의 목적에 따라 형상모델의 단순화 과정이나 이상화(Idealization) 과정이 요구되며, 이 과정에는 형상모델의 LOD 생성이나 상세화 제거(Detail Removal) 등의 구체적인 오퍼레이션이 필요하다¹¹⁾.

현재까지 삼각 메쉬로부터 프로그래시브 모델을 생성시키는 여러 연구결과가 있었다. Hoppe는^{17,14)} 프로그래시브 삼각메쉬 표현 및 단순화 방법론을 제시하였다. 주된 목적은 초기 메쉬 모델의 형상뿐만 아니라 스칼라 속성값에 의해 정의된 전체모습(Overall Appearance)을 최대한 유지시키는데 있다. 하지만, 단순화 과정은 계산과정이 많이 걸리는 단점이 있다. Garland 등¹⁵⁾은 다각형 모델을 빠르게 단순화시키는 방법론을 제시하였는데, Quadric Matrix을 이용하여 Vertex Contraction을 점진적으로 적용시키는 방법이다. 이 연구의 특징은 단순화 과정이 빠를 뿐만 아니라 단순화된 모델의 질도 뛰어나다. 이러한 일련의 프로그래시브 메쉬 생성과 연관되는 사항은 어떻게 효율적으로 모델을 압축시키고 전송시키느냐는 것이다. Rossignac은¹⁵⁾ Edgebreaker라 불리는 삼각메쉬 압축 방법론을 제시하였다. 이 방법론은 단순할 뿐만 아니라 압축률도 뛰어나 많은 곳에서 이용되고 있다.

위의 기존연구 결과들을 살펴보면 주로 삼각메쉬의 프로그래시브 모델 생성 혹은 압축과 관련된 내용이다. 반면에 제품설계에 널리 사용되는 솔리드 모델로부터 프로그래시브 모델 혹은 압축과 관련된 연구결과는 드물다. 이러한 중요성을 언급한 관련내용을 살펴보면, Wu 등은¹⁸⁾ 협업분산환경하에서 솔리드 모델링 과정에서 발생하는 수정을 효과적으로 수행 및 전송을 위한 방법론을 제시하였다. Belaziz 등¹⁹⁾은 디자인 프로세스상에서 분석 틀과의 효과적인 통합을 도울 수 있는 Morphological 분석 방법을 제시하였다. 최동혁 등은²⁰⁾ 특징형상 모델링 히스토리를 재매치시킴으로써 B-Rep의 다중해상도 생성을 위한 방법론을 제시하였다. 재배치 과정에서는 불리언 작업의 합집합과 차집합의 교환과정이 경우에 따라 필요하다. 하지만, 재배치과정 및 각 레벨간의 전이과정에는 불리언 연산이 필요하여 많은 계산시간을 요구한다. 이상현^{11,20)} 등은 불리언 작업 중 합집합과 차집합 사이에는 일반적으로 교환과정이 성립하지 않음을 지적하고, 쉘 구조를 갖는 비다양체 모델을 다중해상도 구현에 적용시켜, 합집합과 차집합 작업사이에 상호교환 법칙이 성립되는 선택적 불리언 작업을 제시하였다.

이재열¹³⁾ 등은 솔리드 모델을 인터넷 상에서 전송시키기 위해서 프로그래시브 솔리드 모델을 제시하였으며, 이는 셀룰러 토폴로지를 이용하여 어떻게 효율적으로 프로그래시브 솔리드 모델을 표현하고 생성할 것인지에 관한 방법을 제시하였다. 일반적으로, 올바른 프로그래시브 모델을 생성하기 위해서는 효율적인 인터넷 전송 자원(Efficient Transmission), 불리언 연산과 같은 비효율적인 작업 최소화(Computational Efficiency), 각 레벨간의 전이 용이(Easy Transition between LODs) 및 LOD의 효율적인 저장(Efficient Storage of LOD), 전송 및 복원 후 정보손실 최소화 등이 요구된다^{7,13,19,21)}. 따라서, 본 논문에서는 이러한 요구를 만족시킬 수 있는 새로운 방법론을 제시하는데 주 목적이 있다.

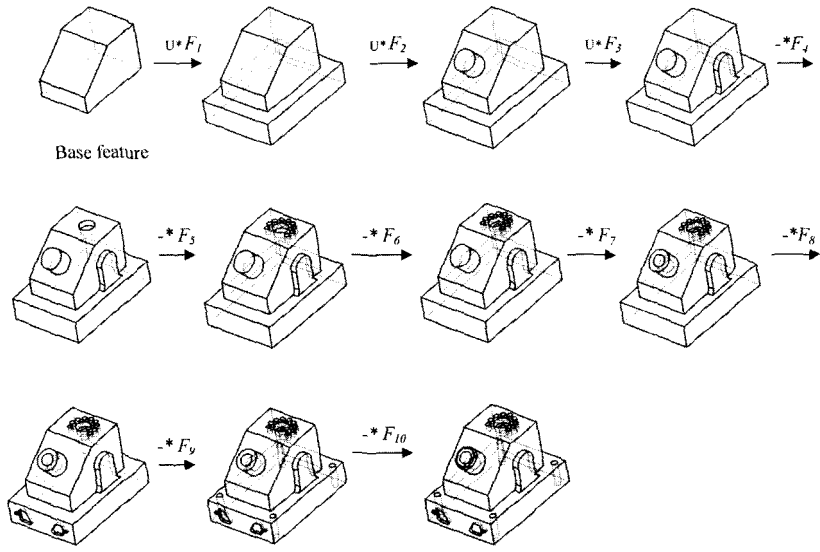
본 논문은 분산환경하에서 효율적인 설계모델 공유 및 전송을 위한 셀룰러 토폴로지 기반 프로그래시브 솔리드모델 표현 및 생성 방법론을 제시한다. 특히 제시된 방법은 특징형상 모델을 기반으로 하며, 이로부터 프로그래시브 솔리드 모델을 생성시키고 이를 인터넷 상에서 효율적으로 전송시키는 방법론을 제시한다. 즉, 디자인 특징형상 집합(Set)으로 설계된 솔리드 모델 SM은 단순화된 초기 프로그래시브 모델 SM⁰와 일련의 n개의 구체적인 상세 변위모델로 변환된다. 일련의 상세변위모델들이 SM⁰에 적용되어 점진적으로 최종모델 SM을 생성시킨다. 또한, 상세변위모델을 SM⁰에 적용시켰을 경우 프로그래시브 솔리드 모델(PSM) 표현은 SM⁰, SM¹, ..., SMⁿ 순으로 모델의 정확성이 높아지며 이러한 사실은 사용자의 목적이나 보안에 따라 올바른 LOD(Level-of-Detail)를 제공할 수 있다. 따라서, 본 연구에서 제시하는 방법론은 삼각메쉬가 아닌 솔리드 모델로부터 프로그래시브 모델을 생성시킬 수 있는 방법이다.

본 논문의 구성은 다음과 같다. 제 2장은 프로그래시브 솔리드 모델을 효율적으로 생성시키기 위한 형상표현 방법에 대해서 설명하고, 제 3장은 2장에서 언급된 형상표현을 기반으로 특징형상모델로부터 프로그래시브 솔리드 모델을 생성시키는 방법에 대해서 기술한다. 제 4장은 구현결과를 보여주며, 마지막으로 5장에서 결론을 맺는다.

2. Progressive Solid Model Representation

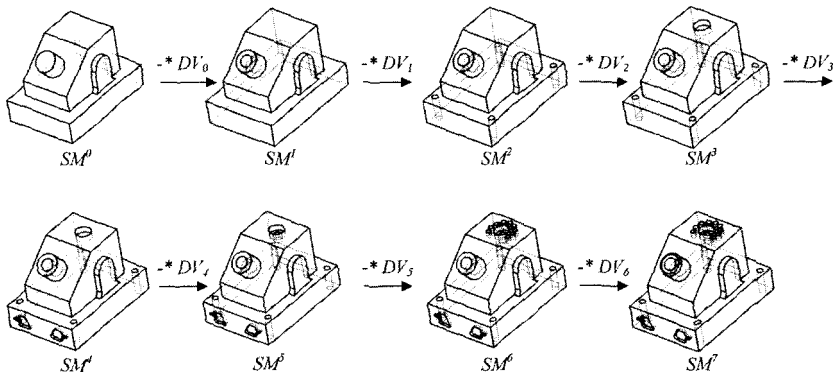
2.1 Overview

특징형상 모델은 초기모델에 특징형상모델들을 주



- 1) U* and -* represent a regularized union and difference operations, respectively.
- 2) F5 is a negative pattern feature. F10 is a transition feature (e.g., rounding).

(a) Feature-based design sequence of a CAD*I part



(b) Progressive solid LOD generation

Fig. 1. Feature-based part design and its corresponding progressive LOD generation.

가함으로써 쉽게 만들어진다. Fig. 1에서 볼 수 있듯이 CAD*I 테스트 파트는 Additive, Negative, Transition 특징형상으로 설계될 수 있다. 일반적으로, 설계된 파트의 디테일 모델은 특정 응용분야에 항상 적합한 것은 아니다. 경우에 따라서는 디테일 모델이 단순화되거나 이상화되어야만 특정 응용분야에 사용될 수 있다. 이러한 단순화 및 이상화 과정 중의 하나인 프로그레시브 모델 생성이 적용되면 사용자는 인터넷 상에서 초기 단순화 모델 SM^0 를 우선적으로 받고 n 상세변위모델($DV_0 \sim DV_{n-1}$)을 점진적으로 받으면

서 최종모델 SM^n 로 디테일화시킬 수 있다. 각각의 상세변위모델 DV_n 는 델타볼륨(Delta Volume)으로 칭하며 이는 SM^i 에서 SM^{i+1} 로 변환시킨다($SM^{i+1} = SM^i - * DV_i$). Fig. 1(b)는 Fig. 1(a)의 특징형상 모델로부터 제시된 방법을 적용시켰을 경우 생성된 프로그레시브 솔리드 모델을 보여준다. 이러한 과정은 분산환경하에서 형상모델 가시화나 협업과정에서 많은 도움을 줄 수 있다. 또한, 사용자의 역할에 따라서 특정한 모델 SM^i 만 접근허용을 가능케 하여 보다 폭넓게 사용될 수 있는 장점이 있다.

본 논문에서는 최종모델 $SM = SM^0$ 으로부터 아래와 같이 SM^0 와 n 상세면위모델을 생성시킨다.

$$(SM=SM^0) \xrightarrow{U^*DV_{n-1}} \dots \xrightarrow{U^*DV_1} SM^1 \xrightarrow{U^*DV_0} SM^0$$

여기서 U^* 는 볼리언 유니온 오퍼레이션이며, DV_i 는 SM^{i+1} 와 SM^i 사이의 델타블륨이다.

위에서 생성된 역변환 과정이 프로그레시브 모델 생성과정이다. 즉,

$$SM^0 \xrightarrow{-^*DV_0} SM^1 \xrightarrow{-^*DV_1} SM^2 \dots \xrightarrow{-^*DV_{n-1}} (SM=SM^0)$$

본 논문에서는 $\{SM^0, \{DV^0, DV^1, \dots, DV_{n-1}\}\}$ 을 프로그레시브 솔리드 모델 표현 (Progressive Solid Model Representation)이라고 칭한다. 특히, 디퍼런스 및 유니온 볼리언 과정은 삼각매쉬의 프로그레시브 모델 생성 과정인 *ecol* 및 *vsplit* 과정에 비유된다⁵⁾.

결론적으로 제시된 프로그레시브 모델은 아래와 같은 성질이 있다:

- Completeness:
 $\Delta \subseteq \bigcup DV_i$ where $\Delta = SM^0 - ^*SM^n$
- Presence:
 $\Delta - ^* \bigcup_{DV_i \in DV} \{DV_i\} \neq \emptyset$ and $DV_i \cap SM^0 \neq \emptyset$
- Non-interaction characteristics:
 $DV_i \cap ^*DV_j = \emptyset$ and $DV_i \cap DV_j \neq \emptyset$
- Volume decreasing property:
 $SM^{i-1} \supset SM^i$ for all $i, j = 1, \dots, n$

하지만 델타블륨과 볼리언 연산은 프로그레시브 모델 생성에 여러 가지 단점을 야기시키며, 이를 극복할 수 있는 방안이 필요하다^{6,13,19)}.

- Computational expensiveness: SM^0 에서 SM^{i+1} 로의 전이는 볼리언연산 $SM^i - ^* DV_i$ 를 필요로하기 때문에 델타블륨에 기반한 표현은 효율적이지 못하다. 또한, 클라이언트가 볼리언 연산을 수행시킬 수 있어야 하기 때문에 *thin* 클라이언트가 아닌 *thick* 클라이언트가 된다.
- Storage problem: $\{SM^0, \{DV_0, DV_1, \dots, DV_{n-1}\}\}$ 은 델타블륨 리스트때문에 원래 모델인 SM^0 보다 훨씬 큰 저장 볼륨을 필요로 한다.
- Efficient transmission: 효율적으로 프로그레시브 모델을 지원하기 위해서는 델타블륨의 단점을 극복할 수 있는 새로운 상세면위모델을 표현방법이 필요하다.

이러한 이유 때문에, 본 연구에서는 셀룰러 토폴로지(Cellular Topology) 개념을 도입하여 프로그레시브 모델을 표현하고 생성시키는 방법을 제시한다. 이를 위해 프로그레시브 특징형상(Progressive Feature) 개념을 제시한다.

2.2 Cellular Topology Representation

프로그레시브 솔리드 모델 생성을 효과적으로 수행하기 위한 셀룰러 토폴로지를 기술한다.

기존연구 결과에 따르면¹⁾, 특징형상모델을 셀룰러 모델에서 정의된 셀로 표현했을 경우 여러 가지 효과를 얻을 수 있다. 예를 들면, 가공특성형상 추출을 용이하게 하거나 설계 및 가공을 함께 지원하기 위한 다중뷰를(Multiple View) 쉽게 제공할 수 있다. 따라서, 본 논문에서는 특징형상 정보를 토폴로지 모델에 적용시켜 특징형상을 셀의 결합과 분해로 표현하여 프로그레시브 모델을 효과적으로 생성할 수 있는 방법을 제시한다. 이는 계산의 복잡화, 저장문제, 인터넷 전송 등에 관한 문제를 해결할 수 있다.

셀룰러 모델은 몇 가지 위상학적 특성을 지니고 있다. 이는 하나의 파트를 일련의 Volumetric Quasi-Disjoint 셀로 표현되며, 각 셀은 특징형상의 외부 혹은 내부를 기술한다. Fig. 2에서 볼 수 있듯이 셀룰러 모델의 위상은 비다양체다(Non-manifold). Spatial Enumeration과는 달리, 셀룰러 모델은 형상을 정확히 표현할 수 있다. 또한, 각 셀의 모습은 정형화된 모습이 아닌 다양하게 정의될 수 있다^{2,3,6)}.

예를 들어, ACISTM가 지원하는 셀룰러 모델을 살펴보면, 우선 하나의 LUMP는 CELL로 세분화된다. 각 CELL은 CFACE로 둘러싸여 있으며, 각 CFACE는 한쪽면만을 가리고 있으며 두 개의 CFACE는 하나의 FACE에 의해서 연결된다. 또한, CELL은 차수(Dimension)에 따라서 CELL2D 혹은 CELL3D로 구분된다. 두 개의 CELL3D는 하나의 FACE에 의해서

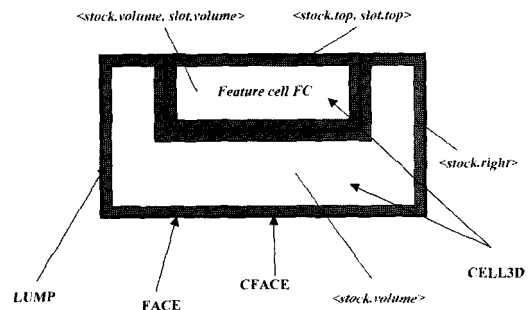


Fig. 2. An example of cell topology and owners¹¹⁾.

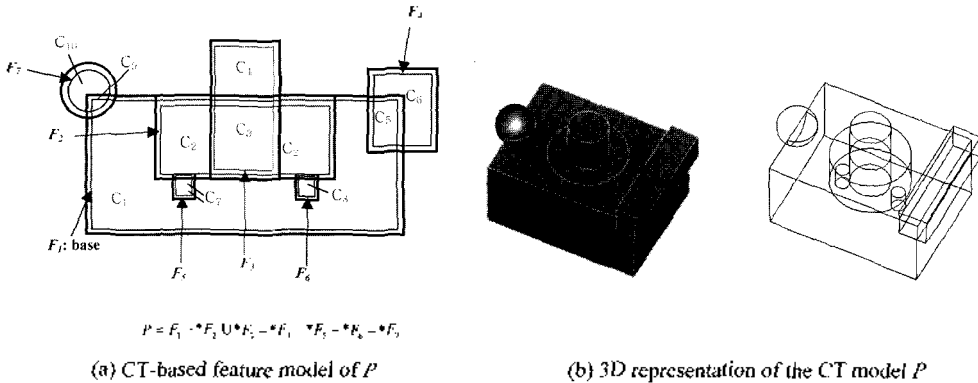


Fig. 3. An example of representing CT-based progressive solid models.

영역이 구분된다. Fig. 3에서 도시된 예는 육면체에서 슬롯을 추가한 후 생성된 셀룰러 모델이다. 결과적으로 2개의 CELL3D가 생성되었으며 이들 사이에는 3개의 FACE에 의해서 영역이 구분된다. 또한, Fig. 2은 각 셀마다의 소유주(Owner) 리스트도 보여준다.

본 논문에서 사용될 용어를 아래와 같이 정의한다.

- 1) 특징형상 셀(Feature Cell) C_k 는 CELL3D 대신 사용됨.
- 2) 특징형상 셀의 최종 소유주는 가장 마지막으로 파트에 추가된 특징형상을 의미함. 예를 들면, Fig. 3에서 FC의 최종 소유주는 슬롯 특징형상임.
- 3) 만약 최종 소유주가 Negative(혹은 Positive) 형상이면, 특징형상 셀도 Negative(혹은 Positive)로 불리며 C_k^- (혹은 C_k^+)로 정의됨.

예를 들면, Fig. 3는 파트 P 가 $P = F_1 - *F_2 U *F_3 - *F_4 - *F_5 - *F_6 - *F_7$ 으로 모델링된 결과로 생성된 CT 모델이다. Fig. 3(a)는 CT 모델을 도시하고, Fig. 3(b)는 CT 모델의 3D 형상을 표현한다.

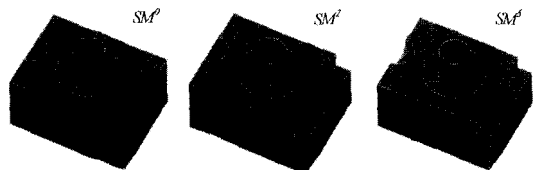
Fig. 4는 Fig. 3에서 정의된 셀룰러 모델의 셀 소유주 및 예상되는 프로그래시브 솔리드 모델을 도시하고 있다. 예를 들면, C_3 의 소유주는 F_1, F_2, F_3 이다. 반대로, 각 특징형상은 여러 셀에 영향을 끼치기도 한다. 즉, F_1 은 $C_1, C_2, C_3, C_5, C_7, C_8, C_9$ 에 영향을 끼친다. 특히, C_6 와 C_{10} 은 더미(Dummy) 셀인데 이들은 Negative 볼륨이며 특징형상 소유주가 하나뿐인 셀로, 프로그래시브 모델 생성에 영향을 끼치지 않는다. Fig. 4(c)는 셀의 병합과 분리에 따른 일련의 프로그래시브 모델의 한 예를 도시하고 있다. 이를 효과적으로 생성시키기 위한 Cut & Paste Operation은 다음 장에서 설명한다.

FEATURE CELL OWNERS	FEATURE CELLS
$C_1 = \{F_1\}$	$F_1 = \{C_1, C_2, C_3, C_5, C_7, C_8, C_9\}$
$C_2 = \{F_1, F_2\}$	$F_2 = \{C_2, C_3\}$
$C_3 = \{F_1, F_2, F_3\}$	$F_3 = \{C_3, C_4\}$
$C_4 = \{F_3\}$	$F_4 = \{C_5, C_6\}$
$C_5 = \{F_1, F_4\}$	$F_5 = \{C_6\}$
$C_6 = \{F_4\}$: dummy cell	$F_6 = \{C_6\}$
$C_7 = \{F_1, F_5\}$	$F_7 = \{C_6\}$
$C_8 = \{F_1, F_6\}$	Dummy cells = $\{C_6, C_{10}\}$
$C_9 = \{F_1, F_5\}$	
$C_{10} = \{F_7\}$: dummy cell	

(a) Cell owner list and feature cells

$SM^0 = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9\}$
$DI^0 = C_5$
$DI^1 = C_3$
$DI^2 = C_4$

(b) Expected progressive solid models SM^i by cell composition and decomposition



(c) Expected CT-based Level-of-Details of the feature model

Fig. 4. Feature cell owners and cell composition.

3. Progressive Solid Model Generation

프로그래시브 솔리드 모델 생성 과정은 초기 특징형상 모델 $SM = SM^0$ 으로부터 SM^i 로 변환시키는 일련의 변환과정이다. 본 연구에서는 파트 $P = SM^0$ 은 F_1, \dots, F_m 의 특징형상으로 설계되었으며, P 는 앞 장에서 설명한 것처럼 CT 모델로 표현이 가능하다고 가정한다. 결과적으로, 모델생성과정을 거치게 되면 프로그

레시브 솔리드 모델 표현 (PSM)은 $\{SM^0, \{PF_0, PF_1, \dots, PF_n\}\}$ (여기서, PF_i 는 레벨 i 에서의 프로그레시브 특징형상)로 표현된다. PSM 생성 프로과정은 우선 SM^0 를 찾고, 1) Negative 특징형상 셀을 찾고, 2) 이들을 클러스터링 하여 분리된 셀 영역(Cell Region) 나누며, 3) 각 셀 영역을 분할하며, 4) 분할된 셀로부터 일련의 프로그레시브 특징형상을 생성시키는 과정을 따른다.

3.1 Progressive Features

본 연구에서는 2장에서 언급된 델타볼륨의 난점을 극복하기 위해서 프로그레시브 특징형상 개념을 제시한다. 이는 셀의 병합과 분할로부터 유도된다. 제시된 개념의 장점은 효율적인 전송 및 저장이 가능하며, LOD간에 진이가 쉽게 이루어진다.

델타볼륨 DV_i 에 대응되는 프로그레시브 특징형상 PF_i 은 속성값을 포함한 델타볼륨의 Face의 Subset으로 정의된다. 즉, PF_i 는 (N_i, O_i, A_i) 로 정의되며, N_i 은 $SM(N_i)$ 에 추가될 Face의 Set이고, O_i 는 $SM(O_i)$ 로부터 제거되어야 할 Face의 Set이며, A_i 은 속성값을 나타낸다.

$$N_i = \partial DV_i - \sum_k f_k, \text{ where } f_k \in \partial DV_i \text{ and}$$

$$f_k \cap \partial SM^i \neq \emptyset$$

$$O_i = A \text{ set of Facelds of } \sum_k f_k,$$

$$\text{where } f_k \in \partial DV_i \text{ and } f_k \cap \partial SM^i \neq \emptyset$$

여기서 ∂SM^i 는 SM^i 를 구성하는 특징형상 셀의 외부 면을 나타내며, f_k 는 DV_i 의 면이다. 또한, 속성값은 색깔, 텍스처, 엔지니어링 정보 등을 포함할 수 있다.

면 f_k 의 Faceld는 다음과 같이 정의된다:

$$\text{Faceld}(f_k) = \{\text{ProgressiveFeatureId}, \text{faceIndex } k, \text{ surfaceType}\}$$

3.2 Constructing SM^0

SM^0 는 너미셀을 제외한 모든 셀의 결합으로 정의된다. 이것은 Positive 특징형상의 모든 셀들을 결합한 것과 같다. 이러한 정의는 SM^0 가 SM^i 의 개략적인 형상모습을 하며, 또한 초기의 단순화된 모델을 정의해야 하기 때문이다. 3.4장에서 SM^0 의 확장에 관해서

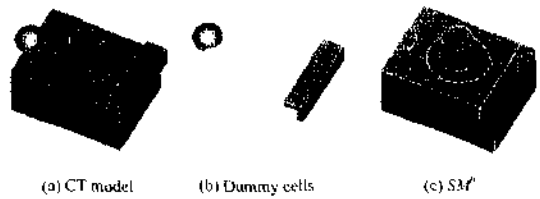


Fig. 5. Constructing SM^0 .

좀더 언급한다. 본 연구에서는, SM^0 로부터 델타볼륨의 Volume Cutting 속성을 유지하도록 정의한다:

$$SM^0 = \bigcup_k *C_k \text{ where the owner of } C_k \text{ is a positive feature.}$$

Fig. 5은 Fig. 3의 CT 모델로부터 정의된 SM^0 를 도시한다.

3.3 Generating Progressive Features

프로그레시브 특징형상 모델을 생성하는 것은 다음의 4 계를 거친다.

- 1) Extracting negative feature cells whose last owners are negative features,
- 2) Clustering them into disjoint cell regions by analyzing the connectivity of the cells,
- 3) Dccomposing each cell region into progressive features and sequencing them,
- 4) Ordering clusters and sequencing progressive features.

3.3.1 Extracting negative feature cells

Negative 특징형상 셀을 찾는 것은 각 셀의 속성값을 비교함으로써 쉽게 찾을 수 있다. 즉,

$$NF = \sum_k C_k^- \text{ where } C_k^- \text{ is not a dummy cell}$$

Fig. 6은 Fig. 5으로부터 추출된 Negative 특징형상들이다.

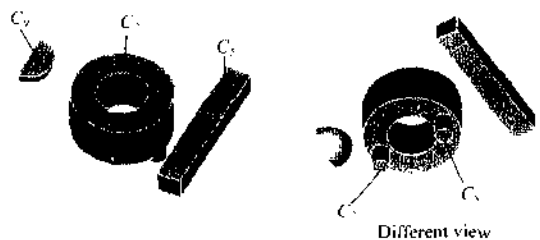


Fig. 6. Negative feature cells.

3.3.2 Clustering feature cells

클러스터링 단계는 Interacting Feature Cell Graph (IFCG)를 작성하여 추출된 Negative 특징형상 셀로부터 분할된 셀 영역(Disjoint Cell Region)을 찾는 과정이다. IFCG에서 각 노드는 특징형상 셀을 에지는 간섭관계를 나타낸다. C_7 와 C_8 의 관계는 공유하는 CFACE의 유무에 의해서 결정된다. 최종 IFCG에는 여러 개의 분할 서브그래프(Disjoint Subgraph)로 구성될 수 있다. 각 서브그래프는 클러스터 (Cluster) 혹은 셀 영역 (Cell Region)으로 불린다. 또한 각 클러스터에 속한 소유주 리스트에 따라 단위(Atomic) 혹은 복합(Composite)로 나뉜다. 만약 모든 특징형상 셀이 하나의 소유주를 가리키면 단위 클러스터이고 그렇지 않으면 복합 클러스터가 된다. Fig. 7은 Fig. 6로부터 유도된 IFCG를 보여주는데, C_2 는 복합이고 나머지는 단위 클러스터다.

3.3.3 Decomposing composite clusters and merging atomic clusters

IFCG에서 특히, 복합 클러스터는 분할과정이 필요하다. 복합 클러스터를 볼륨크기나 특징형상타입과 같은 형상정보에 따라서 단위 클러스터로 분할시킬 수

있다. Fig. 8는 Fig. 7의 $Cluster_2$ 를 분할한 예를 보여준다. 여기서는 특징형상타입에 근거를 두었다. 반대로, 여러 개의 단위 클러스터가 합쳐져서 또 다른 단위 클러스터로 변환될 수 있다.

패턴 특징형상은 복사된 형상의 수 만큼 많은 분할된 형상이 생긴다. 이러한 경우는 속성값을 참조하여 하나의 단위 클러스터로 정의할 수 있다. 예를 들면, C_7 과 C_8 이 패턴형상으로 정의되었다면 하나의 단위 클러스터로 정의가 가능하다.

3.3.4 Ordering clusters and sequencing progressive features

마지막으로, IFCG로부터 프로그레시브 특징형상을 찾아내고 순서를 정해야 한다. 순서는 정해진 기준에 따라 다양한 순서를 생성시킬 수 있다. 정의할 수 있는 기준으로는 형상 볼륨, 접근성 (Accessibility), 뷰 (View Dependency), 저장 크기 등이 있을 수 있다. Fig. 9은 최종적으로 생성된 프로그레시브 모델 및 특징형상을 도시하고 있다. 여기에 이용된 기준은 특징형상 볼륨 및 접근성이다. 노란색으로 표시된 부분은 새로운 프로그레시브 특징형상의 추가되기 전에 삭제되어야 할 면이다.

3.3.5 Progressive transmission

프로그레시브 모델의 용도는 우선 SM^0 를 전송시키고 순차적으로 프로그레시브 특징형상을 전송시키는 것이다. 수신하는 쪽에서 SM^0 로부터 SM^i 를 점진적으로 생성시킬 수 있다. 특히, 프로그레시브 특징형상을 사용함으로써 델타볼륨 사용과 비교해서 보다 효과적인 전송 뿐만 아니라, LOD생성, 또한 각 LOD사이의 효율적인 전이 등이 가능하다. 또한, 업데이트 과정은 불리언 연산과정이 필요없으며, Face Cut & Paste 과정에 의해서 쉽게 이루어질 수 있다.

각 LOD상에서의 전이는 아래와 같이 쉽게 이루어진다:

- Transition from SM^i to SM^{i+1} :

$$\partial SM^{i+1} = \partial SM^i - \partial SM^i(O_{i+1}) + N_{i+1}^{A_{i+1}}$$
- Transition from level SM^{i+1} to SM^i :

$$\partial SM^i = \partial SM^{i+1} + \partial SM^{i+1}(O_{i+1}) - N_{i+1}^{A_{i+1}}$$

여기서 ∂SM^i 는 SM^i 의 Face Set이며, $\partial SM^i(O_{i+1})$ 는 ∂SM^i 로부터 삭제되어야 할 Face Set이고, $N_{i+1}^{A_{i+1}}$ 는 A_{i+1} 속성값을 지니고 ∂SM^i 에 추가되어야 할 Face Set임.

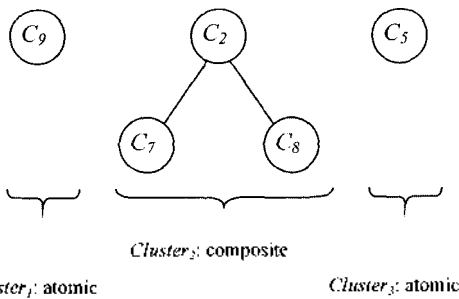


Fig. 7. Feature cell graph and clusters.

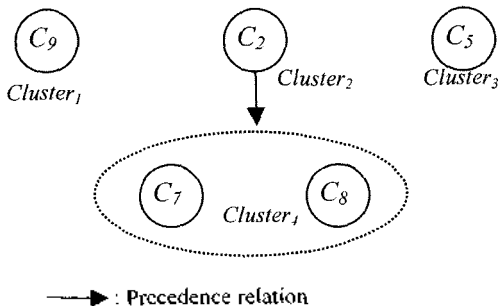


Fig. 8. Decomposition and composition of clusters.

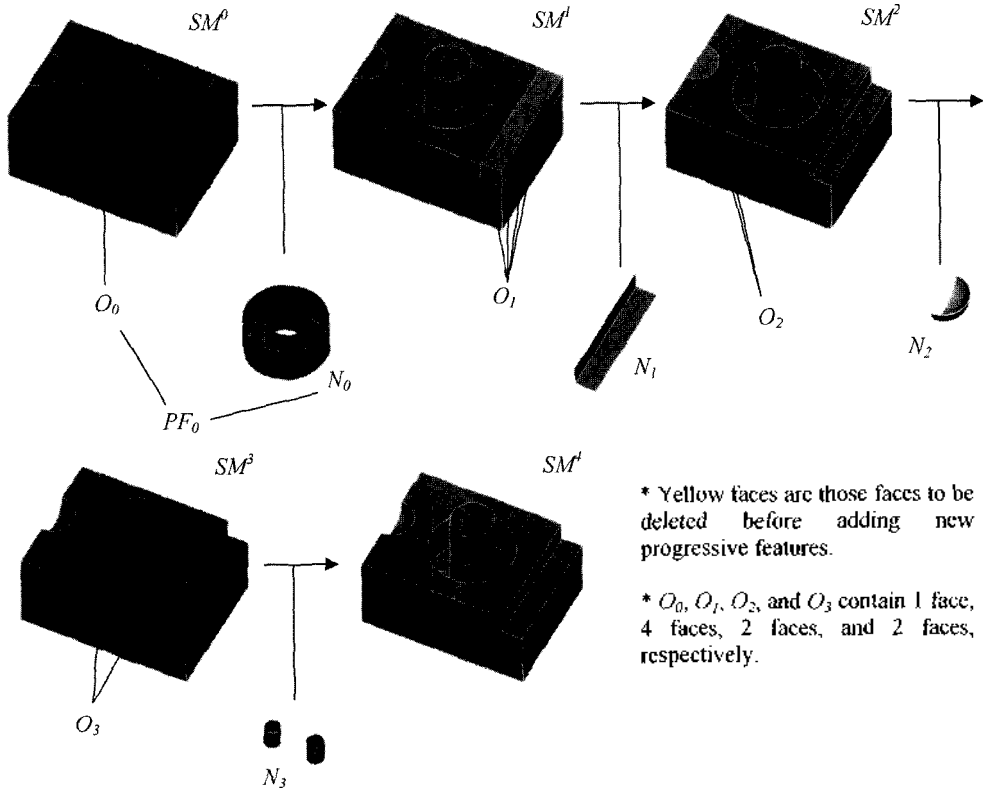


Fig. 9. A sequence of the generated progressive features.

3.4 Extensions

보다 효율적인 프로그레시브 모델을 생성시키기 위해서는 단순히 실제 특징형상 정보에만 의존해서는 안된다. 예를 들면, 모든 특징형상이 Positive 형상이면 프로그레시브 모델을 생성시킬 수 없다. 따라서, 본 연구에서는 이를 극복하기 위해서 특징형상 단면도 배쉬단순화 과정을 거친다^[6].

뿐만 아니라, Positive 특징형상을 Suppression시키는 기능을 부여하고 있다. 즉, Positive 셀의 볼륨이 작을 경우 프로그레시브 모델 초기에 적용시키지 않고 나중에 적용시킴으로써 보다 효과적인 모델을 생성시킬 수 있다. 다음 장의 예에서 이러한 기능의 장점을 볼 수 있다.

4. System Implementation

본 장에서는 솔리드 모델로부터 프로그레시브 모델을 생성한 구현결과를 실예를 통해서 설명한다.

Table 1은 Fig. 9와 10의 예로부터 생성된 프로그레시브 솔리드 모델(SM^0 , SM^1 , SM^2) 및 가시화 모델 사이의 양적인 비교를 보여준다. 솔리드 모델은 ACIS™의 .sat 파일 사이즈이며 가시화 모델은 VRML 파일에 근거를 두었다. Table에서 볼 수 있듯이 인터넷 전송에 따른 프로그레시브 모델의 장점을 볼 수 있다. 특히 Fig. 10에서는 Additive 특징형상을 Suppression시켰을 때 보다 효율적인 프로그레시브 모델을 생성시킬 수 있다는 것을 보여준다. 여기서는 볼

Table 1. Quantitative results of progressive solid models

Models	SM^0		SM^1		SM^2	
	Solid size (KB)	Faceter size (KB)	Solid size	Faceter size	Solid size	Faceter size
Fig. 9	8	22	12	45	20	104
Car in Fig. 10	118	1106	258	1794	684	3394

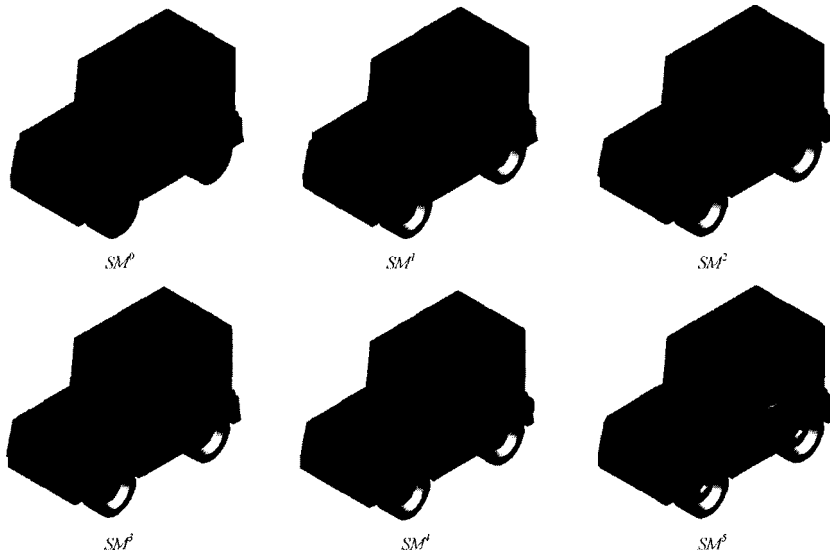


Fig. 10. Progressive solid models of a simplified car.

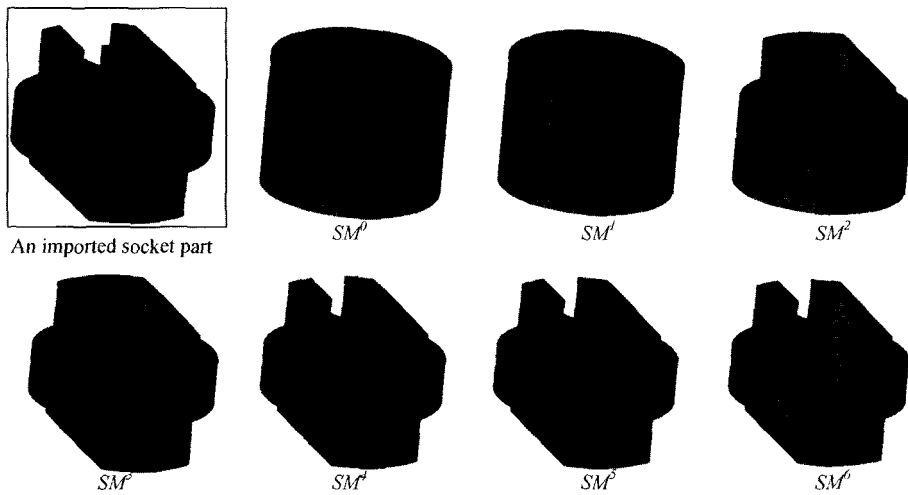


Fig. 11. Progressive solid model generation by feature extraction.

류크기에 기준을 두었다. 자동차의 램프가 여기에 해당된다.

지금까지 도시된 모든 형상은 Design-by-Feature 개념에 근거를 두었다. 디자인 특징형상에 의존적인 것은 기존에 설계된 모델 및 Import된 모델에는 적용될 수 없는 단점이 있다. 이러한 문제점의 해결책 중의 하나는 설계모델로부터 프로그레시브 모델을 추출하는 것이다. Fig. 11의 예를 살펴보면, STEP Translator에 의해서 ACIS로 Import된 모델로서 특징형상 추출 과정 및 프로그레시브 모델 생성과정을 함께 수행한

후 생성된 프로그레시브 솔리드 모델이다. 하지만, 추후 연구가 필요한 분야다.

5. Conclusion

제품 개발 프로세스 상에서 일어나는 단품 설계 및 가상 어셈블리 설계는 삼각메쉬가 아닌 솔리드 모델로 설계된다. 뿐만 아니라, 앞으로, 이러한 제품개발 프로세스가 분산환경하에서 많이 진행되기 때문에 솔리드 모델을 네트워크 상에서 어떻게 효율적으로 공

유 및 교환을 할 것인가가 중요한 이슈로 떠오르고 있다.

본 논문에서는 분산환경하에서 효율적인 설계모델 공유 및 전송을 위한 셀룰러 토폴로지 기반 프로그래시브 솔리드 모델 표현 및 생성 방법론을 제시하였다. 제시된 방법은 인터넷상에서 효과적인 전송을 지원할 뿐만 아니라, 효과적인 LOD 생성 및 각 LOD 사이의 효율적인 전이를 가능케 한다. 특히, 프로그래시브 특징형상 개념을 제시하여 솔리드 모델 전송의 문제점인 불리인 작업을 개선하여 전송 및 복원, 혹은 각 LOD 간의 전이 과정을 프로그래시브 특징형상의 N, O, A를 이용하여 용이하게 지원할 수 있다. 또한, 프로그래시브 특징형상에 속성값을 유지 및 저장시켜 프로그래시브 모델 생성뿐만 아니라 다양한 응용분야에 사용될 수 있도록 기존의 정보를 보존할 수 있다. 하지만, 다음과 같은 후속 연구가 더 필요하다.

- 효율적인 Additive 특징형상의 Suppression 방법: 단순히 특징형상 불필요뿐만 아니라 다양한 기준을 설정하고 이를 바탕으로 효과적인 구현방법론 개발이 필요하다.
- 다양한 순위결정 기준 선상: 프로그래시브 특징형상 순위를 결정하는데 현재는 셀룰러 모델의 쉘의 부피 정보나 Accessibility를 이용하였지만 보다 폭넓은 기준을 적용시킬 필요가 있다.
- 어셈블리 모델로부터 프로그래시브 모델 생성: 다양한 어셈블리 제약조건을 고려하여 어셈블리 모델로부터 프로그래시브 모델을 생성시킬 수 있는 방법론 개발이 필요하다.
- 단순히 특징형상 모델로부터 프로그래시브 모델을 생성시키지 않고 기존의 솔리드 모델로부터 프로그래시브 모델을 효과적으로 추출할 수 있는 방법론 개발이 필요하다.

감사의 글

본 논문은 정보통신부의 “정보통신 선진기반기술개발사업”으로 지원되는 “협업적 제품거래 기술 개발” 과제 결과의 일부입니다.

참고문헌

1. Belaziz, M., Bouras, A. and Burn, J. M., “Morphological analysis for product design,” *Computer-Aided Design*, Vol. 32, No. 5-6, pp. 377-388, 2000.
2. Bidarra, R. and Bronsvoort, W. F., “Semantic feature modeling,” *Computer-Aided Design*, Vol. 32, No. 3, pp. 201-225, 2000.
3. Bidarra, R., de Kraker, K. J. and Bronsvoort, W. F., “Representation and management of feature information in a cellular model,” *Computer-Aided Design*, Vol. 30, No. 4, pp. 301-313, 1998.
4. 최동혁, 김태완, 이진우, “특징형상 변환을 이용한 B-rep 모델의 다중해상도 구현,” 한국 CAD/CAM학회 논문집, 제7권, 제2호, pp. 121-130, 2002.
5. Garland, M. and Heckbert, P., “Surface simplification using quadric error metrics,” *Proceedings of ACM SIGGRAPH '97*, pp. 209-216, 1997.
6. Hoffmann, C. M. and Joan-Arinyo, R., “CAD and the product master model,” *Computer-Aided Design*, Vol. 30, No. 11, pp. 905-918, 1998.
7. Hoppe, H., “Progressive meshes,” *Proceedings of ACM SIGGRAPH '96*, pp. 99-108, 1996.
8. Kim, Y. S., “Recognition of form features using convex decomposition,” *Computer-Aided Design*, Vol. 24, No. 9, pp. 461-466, 1992.
9. Kumer, M. and Gupta, S. K., “A geometric algorithm for automated design of multi-stage molds for manufacturing multi-material objects,” *Proceedings of ACM Solid Modeling '01*, pp. 278-288, 2001.
10. Lee, J. Y. and Kim, K., “A feature-based approach to extracting machining features,” *Computer-Aided Design*, Vol. 30, No. 13, pp. 1019-1035, 1998.
11. Lee, J. Y., Kim, H. and Han, S. B., “Web-enabled feature-based modeling in a distributed design environment,” *Proceedings of ASME/DETC '99*, DFM8941, 1999.
12. Lee, J. Y., Kim, H. and Kim, K., “A web-enabled approach to feature-based modeling in a distributed and collaborative design environment,” *Concurrent Engineering: Research and Applications*, Vol. 9, No. 1, pp. 74-87, 2001.
13. Lee, J. Y., Lee, J.-H., Kim, H. and Kim, H.-S., “Progressive solid models for internet-based design and collaboration,” *Proceedings of ASME/DETC '02*, CIE-34457, 2002.
14. Popovic, J. and Hoppe, H., “Progressive simplicial complexes,” *Proceedings of ACM SIGGRAPH '97*, pp. 217-224, 1997.
15. Rossignac, J., “Edgebreaker: connectivity compression for triangle meshes,” *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No. 1, pp. 41-61, 1999.
16. Rossignac, J. and Requicha, A. A. G., “Constructive non-regularized geometry,” *Computer-Aided Design*, Vol. 23, No. 1, pp. 21-31, 1991.
17. Shah, J. J. and Mantyla, M., *Parametric and Feature-based CAD/CAM*, John Wiley, 1995.
18. Wu, D. and Sarma, R., “Dynamic segmentation and incremental editing of boundary representations in a collaborative design environment,” *Proceedings of ACM Solid Modeling '01*, pp. 289-300, 2001.

- 19. 이상현, 이강수, 박상근, "선택적 볼리안 연산자를 이용한 솔리드 모델의 다중해상도 구현," 한국정밀공학회 춘계학술대회논문집, pp. 833-835, 2002.
- 20. 이상현, 이강수, 박상근, "교환법칙을 만족하는 비다양체 모델의 선택적 볼리안 작업의 개발," 한국정밀

- 공학회 춘계학술대회논문집, pp. 836-839, 2002.
- 21. 이재열, 김 현, 한성배, "네트워크 기반 특징형상 모델링," 한국CAD/CAM학회 논문집, 제5권, 제1호, pp. 12-22, 2000.



이 재 열

1992년 포항공과대학교 산업공학과 학사
 1994년 포항공과대학교 산업공학과 석사
 1998년 포항공과대학교 산업공학과 박사
 1998년~2003년 한국전자통신연구원 분산협업기술연구팀 선임연구원
 2003년~현재 전남대학교 산업공학과 조교수

관심분야: Internet-based CAD and Graphics, Collaborative Virtual Prototyping, Engineering Knowledge Management Using Semantic Web, Collaborative Product Commerce (CPC)



이 주 행

1994년 포항공과대학교 전자계산학과 학사
 1996년 포항공과대학교 전자계산학과 석사
 1999년 포항공과대학교 전자계산학과 박사
 1999년~현재 한국전자통신연구원 인터넷컴퓨팅연구부 분산협업기술연구팀 선임연구원

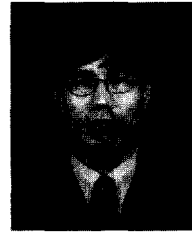
관심분야: Geometric Modeling and Processing, Computer-Aided Design, Computer Graphics, Virtual Reality, Information Visualization, Distributed Computing



김 현

1984년 한양대학교 기계설계학과 학사
 1987년 한양대학교 기계설계학과 석사
 1997년 한양대학교 기계설계학과 박사
 1998년~1999년 한양대학교 산업공학과 겸임교수

1990년~현재 한국전자통신연구원 분산협업기술연구팀장, 책임연구원
 관심분야: Concurrent Engineering, Virtual Engineering, Distributed Collaborative Design, Internet-enabled CAD, Engineering Design Process, Engineering Knowledge Management



김 형 선

1982년 삼지대학교 경영학과 학사
 1992년 광운대학교 컴퓨터공학과 석사
 2001년~현재 대전대학교 컴퓨터공학과 박사과정

1985년~현재 한국전자통신연구원 분산협업기술연구팀 책임연구원
 관심분야: 분산컴퓨팅, Collaborative Product Commerce, 정보보호, 분산데이터베이스