

최대볼륨을 활용한 비교차 볼륨분해

김병철¹ · 문두환^{2†}

¹동아대학교 기계공학과, ²경북대학교 정밀기계공학과

Non-overlapping Volume Decomposition Using Maximum Volumes

Byung Chul Kim¹ and Duhwan Mun^{2†}

¹Department of Mechanical Engineering, Dong-A University

²Department of Precision Mechanical Engineering, Kyungpook National University

Received 17 July, 2013; received in revised form 16 December, 2013; accepted 26 December, 2013

ABSTRACT

In this paper, we propose a non-overlapping volume decomposition method that decomposes a solid model into non-overlapped volumes. The non-overlapping volume decomposition finds non-overlapped volumes from maximum volumes. After introducing the concept of the non-overlapping volume decomposition, we discuss technical issues and solutions for them. The non-overlapping volume decomposition is verified by the experiments with a prototype system. From the experiments, it was found that the non-overlapping volume decomposition shows better result than maximum volume decomposition from the view point of design feature recognition.

Key Words: Design feature recognition, Maximum volume decomposition, Non-overlapping volume decomposition

1. 서 론

볼륨분해(volume decomposition)는 복잡한 형상을 단순한 볼륨으로 분해하는 방법으로, 특징형상 인식^[1-4], CAD 모델 단순화^[5,6], 매시 생성^[7] 등에 응용된다. 이 중, 특징형상 인식은 가장 대표적인 응용 영역이다. 특히, 그래프 기반(graph-based)^[8-11] 및 힌트 기반(hint-based)^[12,13]의 특징형상 인식 방법들이 처리할 수 없는 교차 특징형상을 인식할 수 있다. 특징형상 인식 방법에 대한 비교는 참고 문헌 [14]과 [15]에 잘 정리되어 있다.

볼륨분해는 볼륨을 분해하는 방법에 따라 컨벡스 분해(convex decomposition)^[16,17]와 셀 기반 분해(cell-based decomposition)^[18,19]로 나뉜다. 컨벡스 분해는 원본 형상을 형상의 컨벡스헐(convex hull)과 델타볼륨(delta volume)으로 순차적으로 분해해 나가는 방법이다. 이 방법에서는 분해된 볼륨들의 불리언 합(Boolean union)과 차(Boolean subtraction)의 이진 트리로 원본 형상의 볼륨분해가 표현된다. 그러나 컨벡스 분해는 곡면이 들어간 형상에 대해서는 적용하기 어렵다. 또한 볼륨 분해 시 원본 형상에 없는 새로운 면들이 생성된다.

셀 기반 분해는 원본 형상을 단순한 형상을 가진 셀(cell)로 분해하고, 이 셀들을 큰 볼륨으로 재조합하는 방법이다. 재조합하는 방법에 따라

[†]Corresponding Author, dhmun@knu.ac.kr
©2014 Society of CAD/CAM Engineers

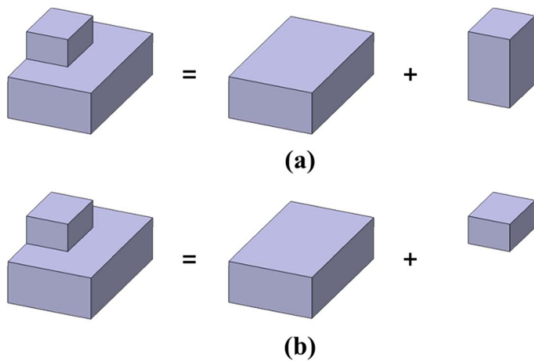


Fig. 1 (a) Maximum volume decomposition, and (b) desirable volume decomposition

다양한 볼륨분해가 가능하다. 대표적인 방법으로는 볼록한 형태의 최대볼륨(maximum volume)으로 분해하는 최대볼륨분해(maximum volume decomposition) 방법이 있다. 그러나 셀 기반 분해는 셀을 조합하는데 걸리는 시간이 조합 복잡도(combinatorial complexity)를 가진다는 단점이 있다. Woo는 이를 해결하기 위해 빠른 셀 기반 분해(fast cell-based decomposition)^[20]를 제안하였다. 김병철은 셀 기반 분해에 랩어라운드(wrap-around)와 볼륨분할(volume split)을 적용한 단계적 볼륨분해(stepwise volume decomposition)^[21]를 제안하였다. 셀 기반 분해는 새로운 면들을 생성하지 않고, 2차곡면까지 처리할 수 있다. 그러나 분해된 볼륨들이 서로 교차하는 문제점이 있다. 이는 가공 특징형상 인식 시에는 크게 문제가 되지 않지만, Fig. 1에서와 같이 설계 특징형상 인식 시에는 부자연스러운 결과를 보여준다.

본 연구에서는 이러한 문제를 해결하기 위해 볼륨들이 최대한 교차하지 않도록 분해하는 방법을 제안한다. Fig. 6(b)와 같이, 일부 경우에는 교차하도록 분해하는 것이 더 자연스럽기 때문에 항상 교차하지 않도록 분해하지는 않는다. 본 연구에서 제안하는 방법은 셀 기반 분해의 한 방법인 최대볼륨분해에 기반을 둔다. 앞에서 언급한 것처럼 셀 기반 분해는 2차곡면 처리가 가능하고, 새로운 면을 생성하지 않기 때문이다.

논문은 다음과 같이 구성되어 있다. 2절에서는 비교차 볼륨분해의 기반이 되는 최대볼륨분해에 대해 설명한다. 3절에서는 비교차 볼륨분해 시 해결해야 할 문제 및 해결방안에 대해 살펴본다. 4절에서는 본 연구에서 제안하는 비교차 볼륨분

해 방법을 설명한다. 5절에서는 테스트 케이스에 대해서 본 연구에서 제안한 방법을 적용한 실험 결과를 살펴보고, 6절에서는 결론 및 향후 연구 방향에 대해서 논의한다.

2. 최대볼륨분해

최대볼륨분해는 형상을 최대볼륨이라고 부르는 단순한 볼륨으로 분해한다. 볼륨 V 가 다음 조건을 만족할 때, V 를 형상 S 의 최대 볼륨이라고 한다^[22,23].

- ① $V \subseteq S$
- ② V 는 오목한 모서리(concave edge)를 가지지 않는다.
- ③ V 의 모든 반공간(halfspace)은 S 의 반공간(halfspace)이다.
- ④ 위의 조건을 만족하는 다른 볼륨 B 에 대해 $B \not\subset V$

최대볼륨분해를 이용해 원본 형상 S 를 최대볼륨으로 분해하는 방법은 다음과 같다.

- ① 원본 형상 S 의 모든 오목한 모서리 $\{e_i\}$ 를 찾는다(Fig. 2(a)).
- ② 각각의 $\{e_i\}$ 에 대해, $\{e_i\}$ 를 공유하는 두 면 f_{i1}, f_{i2} 를 찾아 확장한다(Fig. 2(b)).
- ③ 확장된 $\{f_{i1}, f_{i2}\}$ 과 S 의 비정규화된 볼리언 합(non-regularized Boolean union)을 구하면, S 가 셀들 $\{C_j\}$ 로 분해된다(Fig. 2(c)).

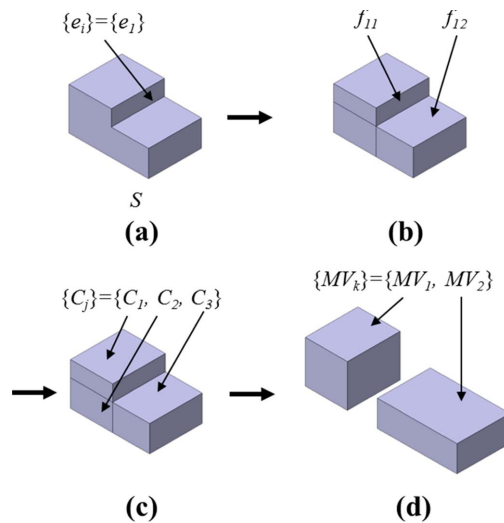


Fig. 2 Concept of maximum volume decomposition

- ④ $\{C_j\}$ 를 최대 볼륨 조건을 만족하도록 조합하여, 최대볼륨 $\{MV_k\}$ 를 찾는다(Fig. 2(d)).

Fig. 2(a)에서는 오목한 모서리 e_l 을 찾고, Fig. 2(b)에서는 이를 공유하는 면 f_{l1} 과 f_{l2} 를 확장하였다. 그 결과 Fig. 2(c)에서 세 개의 셀 C_1, C_2, C_3 가 생성되었고, Fig. 2(d)에서와 같이 C_1 과 C_2, C_2 와 C_3 를 각각 조합하여 최대볼륨 MV_1 과 MV_2 를 찾았다.

3. 비교차 볼륨분해의 기술적 이슈

최대볼륨분해를 이용해 형상을 분해하면 최대 볼륨 간에 겹치는 부분이 생긴다. 예를 들면, Fig. 2(d)의 경우 셀 C_2 가 최대볼륨 MV_1 과 MV_2 에 모두 포함되어 겹치게 된다. 그러나 설계 특정형상의 경우에는 Fig. 1(b)에서와 같이 겹치지 않도록 할 필요가 있다. 본 연구에서는 여러 개의 최대볼륨에 중복되어 포함되는 셀이 한 개의 최대볼륨에만 포함되도록 하여 문제를 해결하고자 한다. 본 연구에서는 최대볼륨분해 이후에 셀이 중복되지 않도록 재조정 된 후의 볼륨들을 비교차 최대볼륨(non-overlapped maximum volume)이라고 부른다.

Fig. 2의 예제에서 비교차 최대볼륨은 Fig. 3에서와 같이 두 가지 경우가 가능하다. 설계자는 Fig. 2(a)의 형상을 모델링 할 때, 설계의도에 따라 Fig. 3(a)와 Fig. 3(b)의 두 가지 방법 중 한 가지를 선택할 수 있기 때문에, 비교차 볼륨분해 결과는 유일하지 않다. 따라서 설계의도를 고려하여 겹치는 셀을 어떤 최대볼륨이 가져갈지를 결정해야 한다.

3.1 최대볼륨 우선순위 결정

형상으로부터 설계의도를 유추하는 것은 어려운 문제이기 때문에, 본 연구에서는 휴리스틱

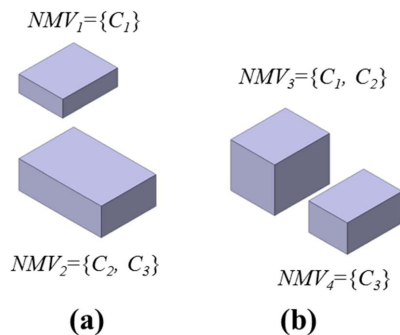


Fig. 3 Non-overlapped maximum volumes

(heuristics)을 적용하여 겹치는 셀을 가져갈 최대 볼륨의 우선순위를 결정하였다.

우선순위를 정하기 위해, 본 연구에서는 설계자가 기준평면(base plane)을 정하고 모델링 한다고 가정한다. 기준평면은 모델링 할 때 형상을 가장 먼저 그리기 위한 기준이 되는 평면이다. 예를 들면, Fig. 4(a)의 형상을 모델링 할 때, 설계자는 가장 아래쪽 면을 기준으로 정해, 기준평면에 가까운 형상을 먼저 모델링 한다. 기준평면은 평면 위의 한 점과 직교하는 두 축으로 정의되며, 이를 각

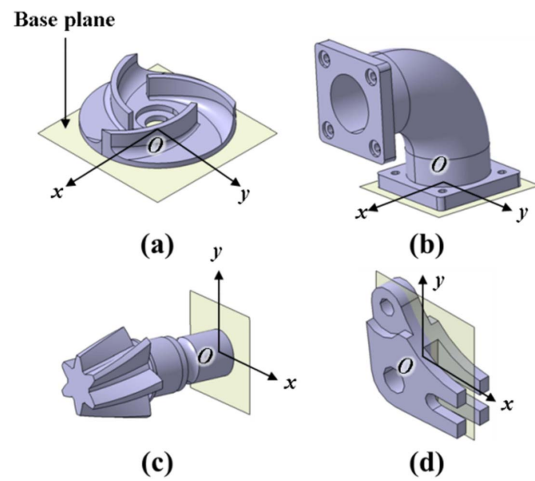


Fig. 4 Examples of the base plane

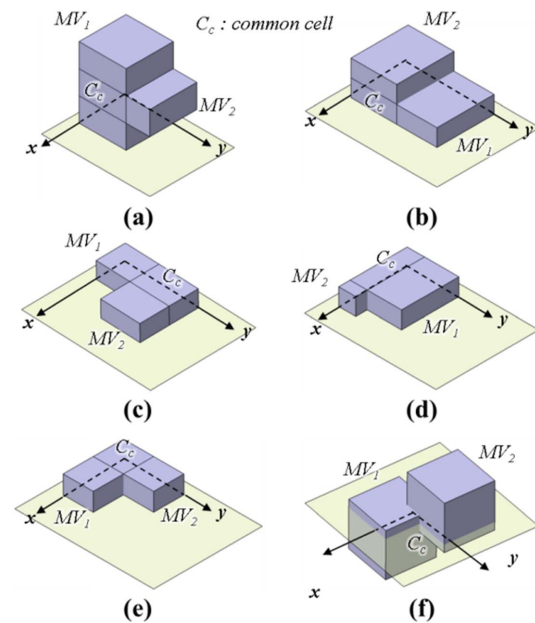


Fig. 5 Examples of the heuristics

각 원점(O 로 표시), x -축, y -축으로 표시한다. Fig. 4는 다양한 기준평면의 예를 보여준다.

기준평면이 정해지면 다음과 같은 순서로 최대볼륨의 우선순위를 정한다.

- ① 기준평면에 거리가 가까운 최대볼륨이 우선 순위가 높다(Fig. 5(a)).
- ② 기준평면에서 z -축 방향을 따라 최대볼륨의 가장 먼 곳이 기준평면에 가까울수록 우선 순위가 높다(Fig. 5(b)).
- ③ 기준평면의 원점에 가까운 최대볼륨이 우선 순위가 높다(Fig. 5(c)).
- ④ 부피가 큰 최대볼륨이 우선 순위가 높다(Fig. 5(d)).
- ⑤ 최대볼륨이 x -축 방향으로 길수록 우선 순위가 높다(Fig. 5(e)).
- ⑥ 최대볼륨이 y -축 방향으로 길수록 우선 순위가 높다.

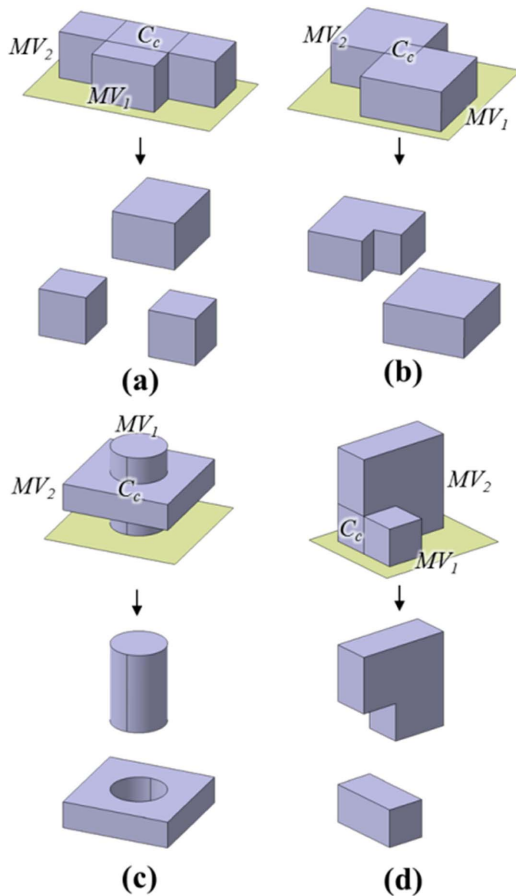


Fig. 6 Exceptional cases of the heuristics

우선순위는 최대볼륨이 기준평면에 가깝고 넓고 평평할수록 높다. 일반적으로 기준평면 위에 넓고 큰 형상을 먼저 생성하기 때문이다. 만약 위의 기준으로도 우선순위가 정해지지 않으면 최대볼륨의 우선순위는 임의로 정한다. 그러나 이러한 경우는 주로 Fig. 5(f)와 같이 원점에 대칭인 경우이므로, 우선순위가 크게 의미가 없어진다.

3.2 예외 경우 처리

최대볼륨의 우선순위를 결정한 후, 겹치는 셀을 우선순위가 높은 최대볼륨에서 가져갈 때 Fig. 6과 같은 경우들이 발생한다.

- Fig. 6(a)는 최대볼륨이 2개이지만, 비교차 최대볼륨이 3개가 된 경우이다. 이는 겹치는 셀 C_c 를 MV_1 이 가져갈 때, MV_2 가 두 개의 볼륨으로 분리됐기 때문이다. 이런 경우에는 Fig. 7(a)와 같이, MV_2 가 C_c 를 가져가야 한다.
- Fig. 6(b)는 겹치는 셀 C_c 를 MV_1 이 가져갈 때, MV_2 가 오목해지는 경우이다. 이 경우에는 C_c 를 MV_2 가 가져간다면 반대로 MV_1 이 오목해진다. 이런 경우에는 Fig. 7(b)와 같이, MV_1 과 MV_2 가 모두 C_c 를 가져가야 한다.
- Fig. 6(c)는 겹치는 셀 C_c 를 MV_1 이 가져갈 때, MV_2 에 구멍이 증가하는 경우이다. 이런 경우에는 보통 MV_2 가 C_c 를 가져가야 한다. 그러나 Fig. 7(c)의 경우, MV_2 가 C_c 를 가져갈 때, MV_1 이 두 개의 볼륨으로 분리되므로, MV_1 과 MV_2

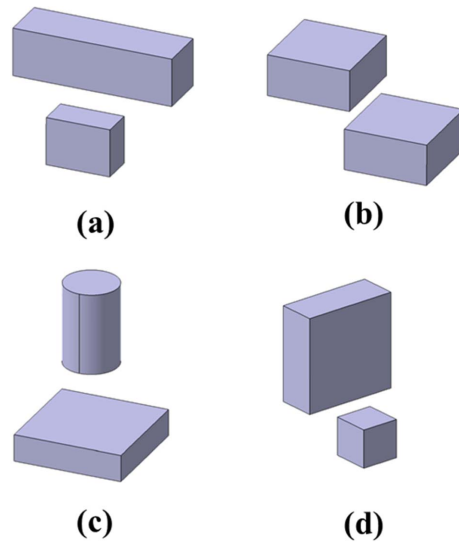


Fig. 7 Corrected non-overlapped maximum volumes for the exceptional cases

모두 C_c 를 가져가야 한다.

- Fig. 6(d)는 겹치는 셀 C_c 을 MV_1 이 가져갈 때, MV_2 가 오목해지는 경우이다. 이런 경우에는 Fig. 7(d)와 같이, MV_2 가 C_c 를 가져가야 한다.

4. 비교차 볼륨분해 방법

형상 S 와 기준평면 BP 가 주어졌을 때, 비교차 볼륨분해를 찾는 알고리즘은 다음과 같다.

- ① 최대볼륨분해를 이용해 S 의 모든 셀 $C=\{C_0, \dots, C_{n-1}\}$ 과 최대볼륨 $MV=\{MV_0, \dots, MV_{m-1}\}$ 을 모두 찾는다. 이 때, 각각의 $MV_j \in MV$ 는 C 의 부분집합으로 구성된다.
- ② 각각의 셀 $C_i \in C(i=0, \dots, n-1)$ 에 대해, C_i 를 포함하는 모든 최대볼륨을 찾는다. 이를 순서가 있는 집합 CMV_i 로 표시한다. 그리고 CMV_i 를 포함하는 순서가 있는 집합을 CMV 라고 하자.
- ③ CMV 의 원소를 $n(CMV_i)$ 를 기준으로 오름차순 정렬한다. 동일한 셀이 세 개 이상의 최대볼륨에 포함되는 경우 중복되어 처리되는 경우가 있다. 이 경우 올바른 결과가 나오도록 하려면, 중복이 적은 셀부터 처리해야 한다. 따라서 정렬을 통해 처리 순서를 정한다.
- ④ 각각의 $CMV_i \in CMV(i=0, \dots, n-1)$ 에 대해, CMV_i 의 원소, 즉 최대볼륨을 BP 를 기준으로 3.1절의 우선순위에 따라 정렬한다. 이는 C_i 를 포함하는 최대볼륨들 간에 우선순위를 결정하는 작업이다. 즉, 높은 최대볼륨이 앞에 오도록 한다.
- ⑤ 각각의 $CMV_i \in CMV(i=0, \dots, n-1)$ 에 대해, 셀 C_i 와 함께 처리해야 할 셀들을 찾는다. 이를 GC_i 로 표시한다. 두 최대볼륨이 두 개 이상의 셀을 공유하는 경우가 있기 때문에, 이 경우 공유하는 모든 셀을 함께 처리해 주어야 한다.
- ⑥ 각각의 셀 $C_i \in C(i=0, \dots, n-1)$ 에 대해, $n(CMV_i) > 1$ 인 경우에 대해 다음을 수행한다.
 - ⑥-1. CMV_i 의 모든 원소들을 이용해, C_i 를 포함하는 최대볼륨의 모든 조합을 찾는다. 이 때, 조합의 우선순위는 3.1절의 우선순위를 따른다.
 - ⑥-2. 앞 단계에서 찾은 각각의 조합 중, 3.2

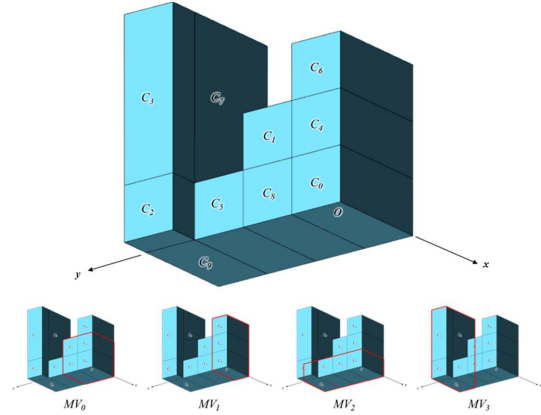


Fig. 8 Example model for explaining the non-overlapping volume decomposition algorithm

절의 예외경우가 발생하지 않는 가장 높은 우선순위의 조합을 찾는다. 이 조합이 C_i 를 가져갈 최대볼륨들이다.

알고리즘에서 단계 ①의 최대볼륨분해 결과는 최대볼륨의 정의에 의해 유일하다. 단계 ②-⑥에서는 정렬과 휴리스틱에 의한 우선순위화를 통해 결과가 랜덤하게 나오지 않도록 알고리즘을 설계하였다. 따라서 비교차 볼륨분해 알고리즘은 항상 유일한 결과를 준다.

비교차 볼륨분해 알고리즘을 설명하기 위해 Fig. 8의 형상에 적용해 보도록 하겠다.

단계 ①: Fig. 8의 형상에 대해 셀과 최대볼륨을 찾으면 다음과 같다.

$$\begin{aligned}
 C &= \{C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9\} \\
 MV &= \{MV_0, MV_1, MV_2, MV_3\} \\
 MV_0 &= \{C_0, C_1, C_4, C_8\} \\
 MV_1 &= \{C_0, C_4, C_6\} \\
 MV_2 &= \{C_0, C_5, C_8, C_9\} \\
 MV_3 &= \{C_2, C_3, C_7, C_9\}
 \end{aligned}$$

단계 ②: 각각의 셀에 대해 CMV 를 찾으면 다음과 같다.

$$\begin{aligned}
 CMV &= \{CMV_0, CMV_1, \dots, CMV_9\} \\
 CMV_0 &= \{MV_0, MV_1, MV_2\} \\
 CMV_1 &= \{MV_0\} \\
 CMV_2 &= \{MV_3\} \\
 CMV_3 &= \{MV_3\} \\
 CMV_4 &= \{MV_0, MV_1\}
 \end{aligned}$$

$$\begin{aligned} CMV_5 &= \{MV_2\} \\ CMV_6 &= \{MV_1\} \\ CMV_7 &= \{MV_3\} \\ CMV_8 &= \{MV_0, MV_2\} \\ CMV_9 &= \{MV_2, MV_3\} \end{aligned}$$

단계 ③: CMV 를 정렬하면 다음과 같다.

$$CMV = \{CMV_1, CMV_2, CMV_3, CMV_5, CMV_6, CMV_7, CMV_4, CMV_8, CMV_9, CMV_0\}$$

단계 ④: 각각의 $CMV_i(i=0, \dots, 9)$ 에 대해 최대 볼륨들의 우선순위를 정한다. 이 때, 기준평면은 Fig. 8에 x -축, y -축, 원점 O 로 정의된 평면이다. 3.1 절의 기준에 따르면, 우선순위는

$$MV_2 > MV_3 > MV_0 > MV_1$$

이다. 여기서 CMV_i 의 나열 순서는 단계 ③에서 정렬된 순서이다.

$$\begin{aligned} CMV_1 &= \{MV_0\} \\ CMV_2 &= \{MV_3\} \\ CMV_3 &= \{MV_3\} \\ CMV_5 &= \{MV_2\} \\ CMV_6 &= \{MV_1\} \\ CMV_7 &= \{MV_3\} \\ CMV_4 &= \{MV_0, MV_1\} \\ CMV_8 &= \{MV_2, MV_0\} \\ CMV_9 &= \{MV_2, MV_3\} \\ CMV_0 &= \{MV_2, MV_0, MV_1\} \end{aligned}$$

단계 ⑤: 각각의 $CMV_i(i=0, \dots, 9)$ 에 대해, 함께 처리해야 할 셀들을 찾는다. GC_4 와 GC_8 에 대해서만 함께 처리할 셀들이 존재한다.

$$\begin{aligned} GC_4 &= \{C_0\} \\ GC_8 &= \{C_0\} \end{aligned}$$

Fig. 8에서 보면, MV_0 와 MV_1 에 대해서는 C_0 과 C_4 를, MV_0 과 MV_2 에 대해서는 C_0 와 C_8 을 함께 처리해 주어야 한다.

단계 ⑥: 각각의 셀 $C_i(i=0, \dots, 9)$ 에 대해 C_i 를 가져갈 최대볼륨을 결정한다. $CMV_1, CMV_2, CMV_3, CMV_5, CMV_6, CMV_7$ 은 한 개의 원소만을 가지기 때문에 무시한다. 이 경우, C_1 은 MV_0 , C_2 는 MV_3 , C_3 는 MV_3 , C_5 는 MV_2 , C_6 은 MV_1 , C_7 은 MV_3 이 가져간다. 따라서 $CMV_4, CMV_8, CMV_9, CMV_0$ 의 경우

만 고려하면 된다.

CMV_4 의 경우, MV_0 와 MV_1 의 조합을 찾으면 다음과 같은 순서가 된다.

$$\{MV_0\}, \{MV_1\}, \{MV_0, MV_1\}$$

위의 조합 순서에 따라 MV_0 가 C_4 와 함께 처리할 C_0 를 가져가는 경우, MV_0, MV_1 이 모두 3.2절의 예외경우에 해당하지 않는다. 따라서,

$$\begin{aligned} MV_0 &= \{C_0, C_1, C_4, C_8\} \\ MV_1 &= \{C_0\} \\ MV_2 &= \{C_0, C_5, C_8, C_9\} \\ MV_3 &= \{C_2, C_3, C_7, C_9\} \end{aligned}$$

가 된다.

CMV_8 의 경우에도 동일한 방법으로, MV_2 가 C_8 과 C_0 를 가져간다. 따라서,

$$\begin{aligned} MV_0 &= \{C_1, C_4\} \\ MV_1 &= \{C_0\} \\ MV_2 &= \{C_0, C_5, C_8, C_9\} \\ MV_3 &= \{C_2, C_3, C_7, C_9\} \end{aligned}$$

가 된다.

CMV_9 의 경우, MV_2 와 MV_3 의 조합을 찾으면 다음과 같은 순서가 된다.

$$\{MV_2\}, \{MV_3\}, \{MV_2, MV_3\}$$

위의 조합 순서에 따라 MV_2 가 C_9 를 가져가야 하지만, 이 경우 MV_3 가 오목해진다. 다음 순서로 MV_3 가 C_9 를 가져가는 경우에는 문제가 생기지 않으므로, MV_3 가 C_9 를 가져간다. 따라서,

$$\begin{aligned} MV_0 &= \{C_1, C_4\} \\ MV_1 &= \{C_0\} \\ MV_2 &= \{C_0, C_5, C_8\} \\ MV_3 &= \{C_2, C_3, C_7, C_9\} \end{aligned}$$

가 된다.

CMV_0 의 경우에도 C_9 를 MV_2 가 가져간다. 따라서,

$$\begin{aligned} MV_0 &= \{C_1, C_4\} \\ MV_1 &= \{C_0\} \\ MV_2 &= \{C_0, C_5, C_8\} \\ MV_3 &= \{C_2, C_3, C_7, C_9\} \end{aligned}$$

가 된다.

최종적으로 찾은 비교차 최대볼륨은 다음과 같다.

$$NMV_0 = \{C_1, C_4\}$$

$$NMV_1 = \{C_6\}$$

$$NMV_2 = \{C_0, C_5, C_8\}$$

$$NMV_3 = \{C_2, C_3, C_7, C_9\}$$

5. 구현 및 실험

본 연구에서 제안하는 방법에 따라 프로토타입 시스템을 개발하였다. 프로토타입 시스템은 C++ 환경에서 ACIS 형상 모델링 커널과 Hoops3D 가시화 라이브러리, MFC(Microsoft foundation classes)를 이용하여 구현되었다.

비교차 볼륨분해는 먼저 최대볼륨분해를 수행해야 한다. 따라서 최대볼륨분해의 단점인 시간이 오래 걸리는 문제와 볼륨분해 시 감산적 볼륨을 처리할 수 없다는 문제를 가지고 있다. 이를 해결하기 위해, 본 연구의 구현에서는 단계적 볼륨분해^[21]에서와 같이 필렛/라운드/모따기 제거, 래어라운드 연산 및 볼륨분할 연산을 사전에 적용하였다.

Fig. 9는 ANC101 파트에 본 연구에서 제안한 비교차 볼륨분해를 적용한 결과이다. Fig. 9에서 실선으로 표시한 부분이 비교차 볼륨분해가 적용된 부분이다. 비교를 위해 최대볼륨분해를 적용한 결

과가 Fig. 10에 실선으로 표시되어 있다. Fig. 9에서는 파트의 바닥면을 기준평면으로 지정하였다.

Fig. 10을 보면, 분해된 볼륨들이 서로 겹치는 것을 볼 수 있다. 겹치는 부분은 Fig. 10의 가장 왼쪽에 있는 볼륨들에서 두드러진다. 반대로 동일한 볼륨이 Fig. 9에서 보면 겹치지 않게 분해된 것을 확인할 수 있다.

Fig. 11은 블레이드(blade) 파트에 비교차 볼륨분해를 적용한 결과이고, Fig. 12는 비교를 위해 최대볼륨분해를 적용한 결과이다. 비교를 위해 비교차 볼륨분해와 최대볼륨분해의 결과를 다른 볼륨들보다 크게 표현하였다. Fig. 12에서 분해된 블레이드가 Fig. 11에서 분해된 블레이드보다 축방향으로 더 긴 것을 볼 수 있다. 이는 세 개의 블레이드가 원판과 교차되어 분해되었기 때문이다. 반면에, Fig. 11에서 분해된 블레이드는 교차되지 않고 분해되었기 때문에 축방향으로 짧다.

Table 1은 ANC101 파트와 블레이드 파트에 비교차 볼륨분해와 최대볼륨분해를 적용하는 데 걸린 소요시간이다. 이는 인텔 코어 i7 CPU와 8GB RAM을 가진 PC에서 측정되었다. 비교차 볼륨분해는 최대볼륨분해를 기반으로 수행되기 때문에, Table 1에서 최대볼륨분해에 소요된 시간과 비교차 볼륨분해에 소요된 시간 차이가 순수하게

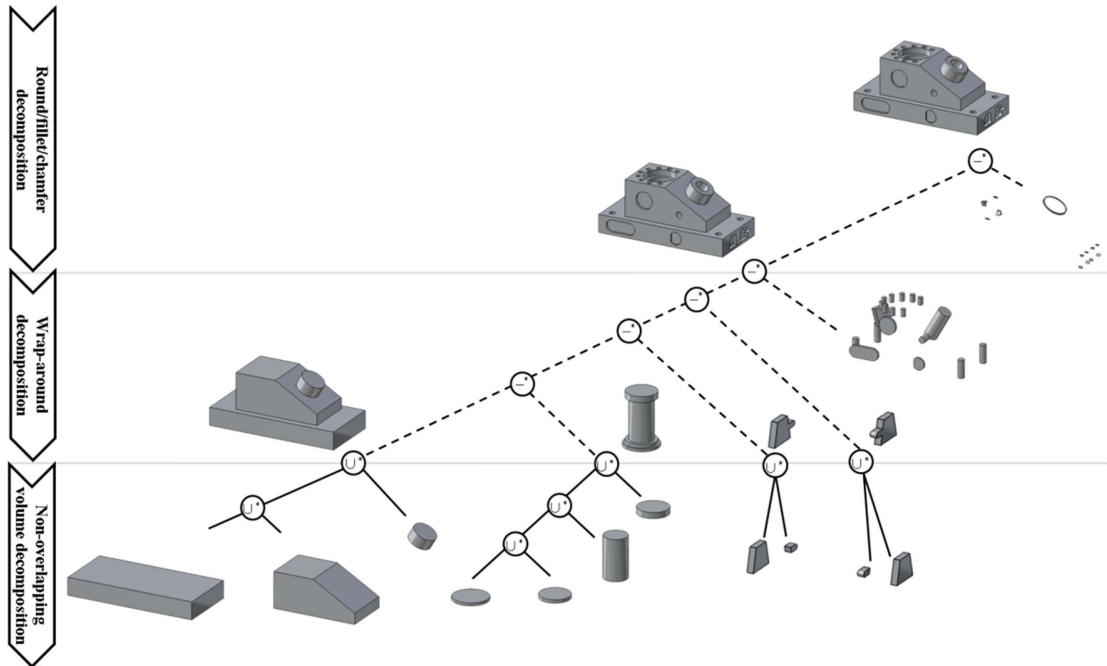


Fig. 9 Non-overlapping volume decomposition of ANC101 part

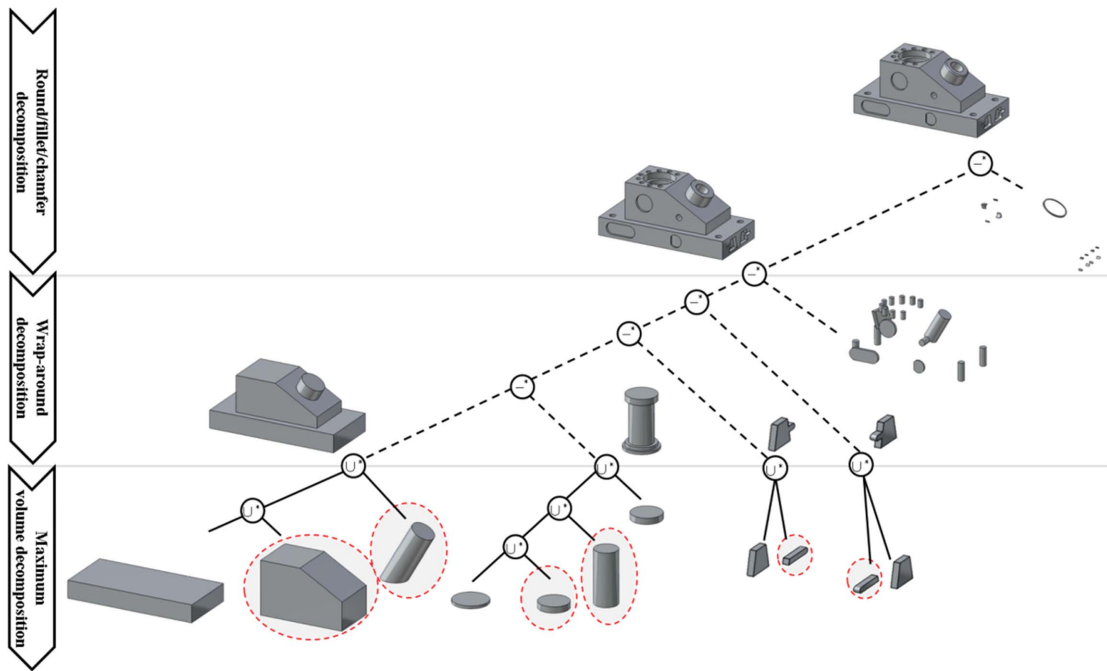


Fig. 10 Maximum volume decomposition of ANC101 part

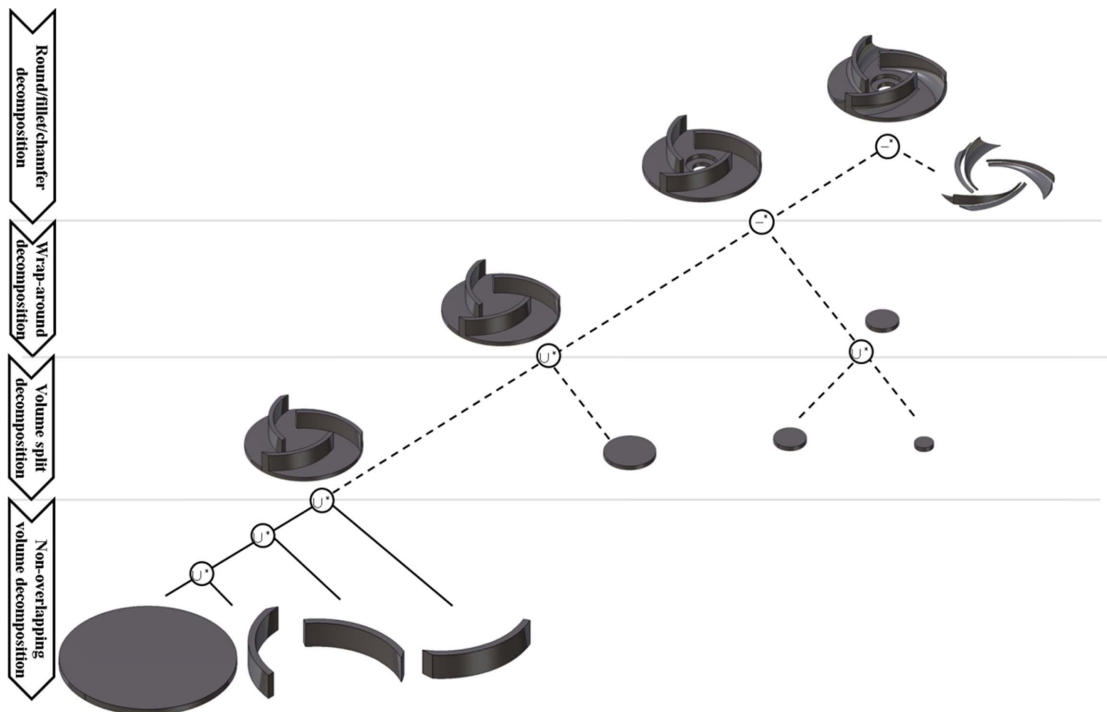


Fig. 11 Non-overlapping volume decomposition of the blade part

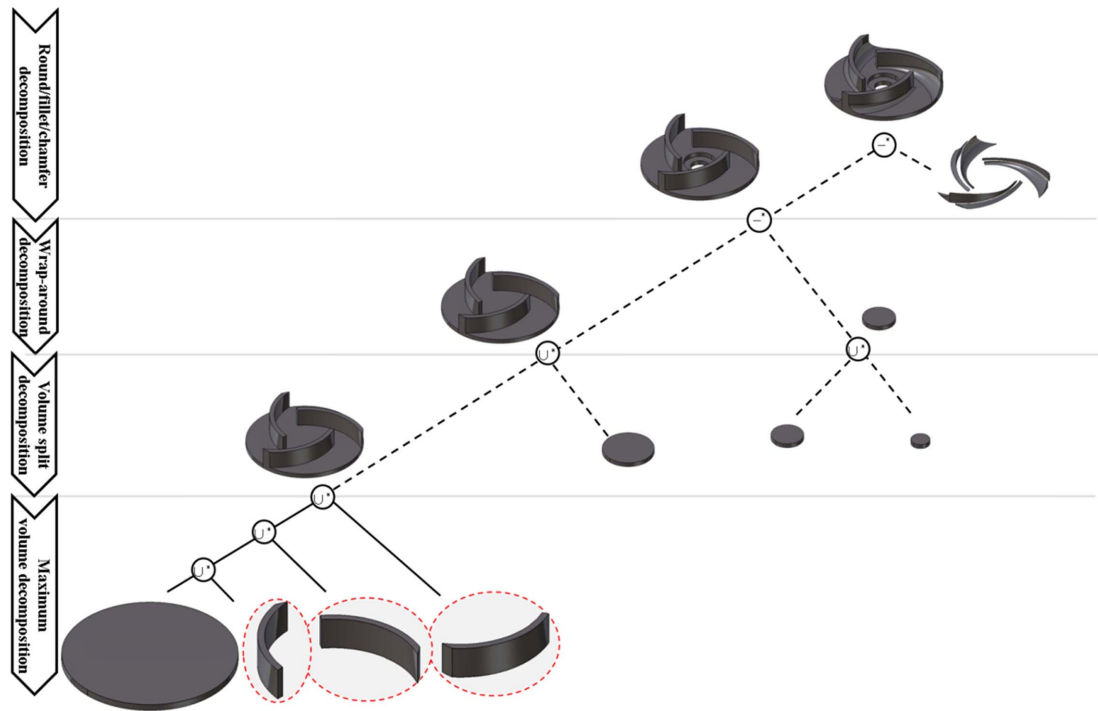


Fig. 12 Maximum volume decomposition of the blade part

Table 1. Decomposition time of each method

	Non-overlapping volume decomposition	Maximum volume decomposition	Net
ANC101	436 ms	458 ms	22 ms
Blade	931 ms	1026 ms	95 ms

비교차 볼륨분해를 계산하는데 걸린 시간이다. 이 시간은 최대볼륨분해 시간에 매우 작은 것을 알 수 있다.

Fig. 13은 비교차 볼륨분해와 단계적 볼륨분해의 볼륨분할 연산^[21]의 결과를 비교한 것이다. 볼륨분할도 형상을 겹치지 않도록 분해하는 방법이지만, 오목한 내부루프(concave inner loop)가 존재하는 부분에만 적용 가능하다. 따라서 Fig. 13(b)와 같이, 오목한 내부루프 형성이 안 될 경우 형상을 겹치지 않도록 분할할 수 없다.

6. 결 론

본 논문에서는 비교차 볼륨분해 방법을 제안하였다. 비교차 볼륨분해는 형상을 서로 겹치지 않

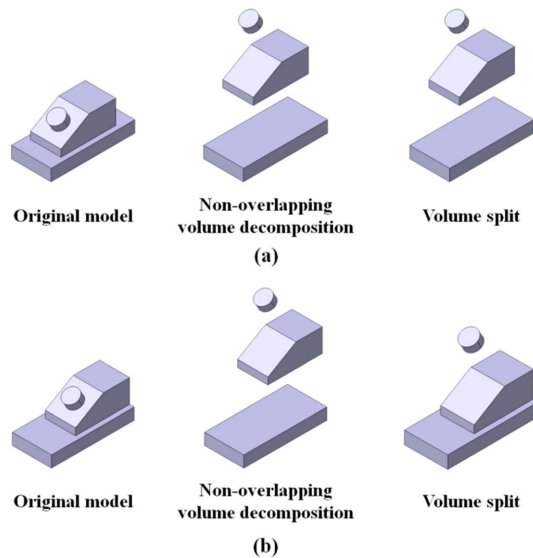


Fig. 13 Comparison of non-overlapping volume decomposition and volume split

는 볼륨으로 분해하는 방법으로, 설계특징형상 인식을 위한 볼륨분해 방법으로 사용될 수 있다. 이 방법은 볼륨 분해 시, 가산적 특징형상 들간에는 교차가 일어나지 않도록 모델링을 하는 기계 3D

CAD 시스템의 특징형상 기반 모델링 관점에서 최대볼륨분해보다 자연스러운 분해결과를 보여준다. 본 논문에서는 비교차 볼륨분해 시 고려해야 할 부분과 알고리즘을 제시하였고, 구현 및 실험을 통해 제안한 방법을 확인하였다.

그러나 비교차 볼륨분해는 최대볼륨분해에 기반을 두고 있기 때문에 분해 시간이 오래 걸린다는 점과 감산적 볼륨을 처리할 수 없다는 점은 동일하다. 이를 해결하기 위해 비교차 볼륨분해를 적용하기 전 랩어라운드 연산과 볼륨분해 연산을 적용할 필요가 있었다.

비교차 볼륨분해는 최대볼륨분해 시 생성된 셀과 최대볼륨을 이용하기 때문에, 최대볼륨의 품질에 영향을 받는다. 따라서 설계특징형상 인식에 좀더 적합하게 볼륨을 분해할 수 있는 방법에 대한 연구가 진행되어야 한다.

감사의 글

이 논문은 동아대학교 교내연구비 지원에 의하여 연구되었음을 밝힙니다.

References

1. Kim, Y.S., 1992, Recognition of Form Features Using Convex Decomposition, *Computer-Aided Design*, 24(9), pp.461-476.
2. Kailash, S., Zhang, Y. and Fuh, J., 2001, A Volume Decomposition Approach to Machining Feature Extraction of Casting and Forging Component, *Computer-Aided Design*, 33(8), pp.605-617.
3. Kim, Y.S. and Wang, E., 2002, Recognition of Machining Features for Cast then Machined Parts, *Computer-Aided Design*, 34(1), pp.71-87.
4. Woo, Y. and Sakurai, H., 2002, Recognition of Maximal Features by Volume Decomposition, *Computer-Aided Design*, 34(3), pp.195-207.
5. Woo, Y., 2007, Decomposition-Based Simplification of Machined Part in Solid Model, *Transactions of the Society of CAD/CAM Engineers*, 12(2), pp.101-108.
6. Woo, Y., 2009, Automatic Simplification of Solid Models for Engineering Analysis Independent of Modeling Sequences, *Journal of Mechanical Science and Technology*, 23(7), pp. 1939-1948.
7. Lu, Y., Gadh, R. and Tautges, T., 2001, Feature Based Hex Meshing Methodology: Feature Recognition and Volume Decomposition, *Computer-Aided Design*, 33(3), pp.221-232.
8. Joshi, S. and Chang, T.C., 1988, Graph Based Heuristics for Recognition of Machined Features from a 3-D Solid Model, *Computer-Aided Design*, 20(2), pp.58-66.
9. Chuang, S.H. and Henderson, M.R., 1990, Three-Dimensional Shape Pattern Recognition Using Vertex Classification and the Vertex-Edge Graph, *Computer-Aided Design*, 22(6), pp.377-387.
10. Gavankar, P. and Henderson, M.R., 1990, Graph-Based Extraction of Protrusions and Depressions from Boundary Representations, *Computer-Aided Design*, 22(7), pp.442-450.
11. Gao, S. and Shah, J., 1998, Automatic Recognition of Interacting Machining Features Based on Minimal Condition Subgraph, *Computer-Aided Design*, 30(9), pp.727-739.
12. Vadenbrande, J.H. and Requicha, A.A.G., 1993, Spatial Reasoning for the Automatic Recognition of Machinable Features in Solid Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12), pp.1269-1285.
13. Regli, W.C., Gupta, S.K. and Nau, D.S., 1995, Extracting Alternative Machining Features: an Algorithmic Approach, *Research in Engineering Design*, 7(3), pp.173-192.
14. Han, J.H., Pratt, M. and Regli, W.C., 2000, Manufacturing Feature Recognition from Solid Models: a Status Report, *IEEE Transactions on Robotics and Automation*, 16(6), pp.782-796.
15. Shah, J.J., Anderson, D., Kim, Y.S. and Joshi, S., 2001, A Discourse on Geometric Feature Recognition from CAD Models, *Journal of Computing and Information Science in Engineering*, 1(1), pp.41-51.
16. Tang, K. and Woo, T., 1991, Algorithmic Aspects of Alternating Sum of Volumes. Part I: Data Structure and Difference Operation, *Computer-Aided Design*, 23(5), pp.357-366.
17. Kim, Y.S. and Wilde, D.J., 1992, A Convergent Convex Decomposition of Polyhedral Objects, *Journal of Mechanical Design*, 114(3), pp.468-477.
18. Sakurai, H., 1995, Volume Decomposition and Feature Recognition, Part I: Polyhedral Objects, *Computer-Aided Design*, 27(11), pp.833-843.
19. Sakurai, H. and Dave, P., 1996, Volume Decomposition and Feature Recognition, Part II: Curved Objects, *Computer-Aided Design*, 28(6-7), pp. 519-537.
20. Woo, Y., 2003, Fast Cell-Based Decomposition and Applications to Solid Modeling, *Computer-Aided Design*, 35(11), pp.969-977.

21. Kim, B. C., Kim, I., Han, S., and Mun, D., 2013, Stepwise Volume Decomposition Considering Design Feature Recognition, *Transactions of the Society of CAD/CAM Engineers*, 18(1), pp.71-82.
22. Woo, Y. and Lee, S.H., 2006, Volumetric Modification of Solid CAD Models Independent of Design Features, *Advances in Engineering Software*, 37(12), pp.826-835.
23. Woo, Y., 2008, Modification of Solid Models Independent of Design Features, *Transactions of the Society of CAD/CAM Engineers*, 13(2), pp.131-138.



김 병 철

2001년 고려대학교 기계공학과 학사
 2003년 한국과학기술원 기계공학과 석사
 2008년 한국과학기술원 기계공학과 박사
 2008년 (주)부품디비 책임연구원
 2009년~2012년 삼성중공업 산업기술연구소 책임연구원
 2012년~현재 동아대학교 기계공학과 조교수
 관심분야: Feature-based and parametric design, CAD data exchange, geometric modeling



문 두 환

1999년 고려대학교 기계공학과 학사
 2001년 한국과학기술원 기계공학과 석사
 2006년 한국과학기술원 기계공학과 박사
 2006년~2010년 한국해양연구원 선임연구원
 2010년~현재 경북대학교 정밀기계공학과 조교수
 관심분야: Computer-aided design, Industrial data standards for product data exchange, Product lifecycle management, Knowledge-based engineering, VR for Engineering Applications