

## 설계 프리미티브 간의 교차형상을 통한 가공 피처 인식

정채봉\*, 김재정\*\*

### Machining Feature Recognition with Intersection Geometry between Design Primitives

Chae Bong Jung\* and Jay Jung Kim\*\*

#### ABSTRACT

Producing the relevant information (features) from the CAD models for CAM, called feature recognition or extraction, is the essential stage for the integration of CAD and CAM. Most feature recognition methods, however, have problems in the recognition of intersecting features because they do not handle the intersection geometry properly. In this paper, we propose a machining feature recognition algorithm, which has a solid model consisting of orthogonal primitives as input. The algorithm calculates candidate features and constitutes the Intersection Geometry Matrix which is necessary to represent the spatial relation of candidate features. Finally, it recognizes machining features from the proposed candidate features dividing and growing algorithms using half space and Boolean operation. The algorithm has the following characteristics: Though the geometry of parts is complex due to the intersections of design primitives, it can recognize the necessary machining features. In addition, it creates the Maximal Feature Volumes independent of the machining sequences at the feature recognition stage so that it can easily accommodate the change of decision criteria of machining orders.

**Key words :** CAD, CAM, Feature recognition, Boolean operation

#### 1. 서 론

CAM을 수행하기 위해서는 주어진 CAD 모델로부터 가공에 필요한 정보(가공피처)를 인식하는 과정이 필요한데, 이는 일반적인 CAD 모델 내에는 가공에 필요한 정보가 포함되어 있지 않고 단지 부품의 형상에 관한 정보만이 존재하기 때문이다.

일반적으로 피처(feature)는 관심 어플리케이션(application), 혹은 도메인(domain) 내에서 의미를 가지는 기술적인 객체(technical object)이다. 좀더 구체적으로 말하면, 엔지니어가 제품(product)에 대해 추론(reasoning)하는 데 유용한 속성(attribute)이나 지식(knowledge) 등을 연상시킬 수 있는 공학적 의미

(engineering significance)를 내포하는 일반적 형태(generic shape)이다<sup>[1]</sup>. 이러한 피처를 CAD모델로부터 추출하는 것을 피처 인식(feature recognition)이라 하며, 일반적으로 가공 피처의 자동 인식을 의미한다. 또한 하나의 부품이 대상 도메인에서 의미를 가지는 피처들로 완전히 표현되었을 때, 이를 그 부품의 피처 모델(feature model)이라 한다. 피처 모델을 만드는 방법은 부품의 솔리드 모델로부터 직접 피처를 추출해 내는 피처 인식(feature recognition) 방법과, 사전에 정의된 피처를 사용하여 모델을 생성하는 피처 기반 모델링(feature based modeling)방법이 있다<sup>[2]</sup>. 그러나 후자 역시 모델링 시에 사용된 피처들이 항상 가공에 의미 있는 가공 피처와 일대일로 대응되지는 않기 때문에, 피처 기반 설계에 의한 피처 모델을 가공 피처 모델로 변환해 주는 작업이 필요하다. 이를 피처 변환(feature conversion), 또는 피

\*학생회원, 한양대학교 기계설계학과

\*\*중신회원, 한양대학교 기계공학부 부교수

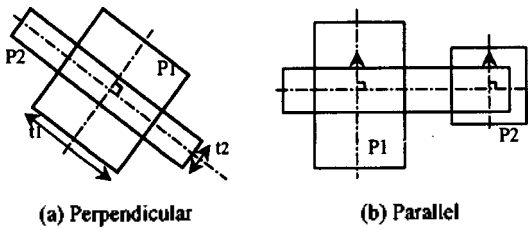


Fig. 1. Orthogonal primitives.

쳐 사상(feature mapping)이라 하며, 여기에서도 피쳐 인식 알고리즘이 기능의 핵심적 역할을 수행하고 있다<sup>13)</sup>.

피쳐 인식에 관한 연구는, 70년대 말 Tony Woo<sup>14)</sup>와 Kyprianou<sup>15)</sup>가 피쳐 인식 분야에 선두적인 연구 논문을 발표한 이래로 현재까지 20여년 동안 많은 연구자들에 의해 발전을 거듭해 왔다. 피쳐 인식을 주기능으로 하는 상용 소프트웨어 제품으로는 Geometric Software(GSSL)사나 Allied-Signal사 등에서 초보적인 제품들이 나오고 있는 실정이며, 이들 제품은 대체로 고립된 형태로 존재하는, 즉 상호 교차 하지 않는 피쳐들만을 인식하고 있으며 입력 부품 CAD데이터의 기하학적 구성도 상당히 제한적이다.

따라서 본 연구에서는 피쳐가 상호 교차하는 경우에도 유용한 피쳐를 인식 할 수 있는 방법을 구현하였다. 그러나 문제를 단순화 하기 위해 비교적 교차 형상이 규칙적인 직교 프리미티브를 갖는 솔리드 모델에 적용되는 가공 피쳐 인식 알고리즘의 개발을 목적으로 하였다. 여기서 직교(Orthogonal)라 함은 Fig. 1에서 보듯이 프리미티브들 간의 중심 축이 서로 평행(parallel)하거나 직각(perpendicular)이며, 각 프리미티브의 두께가 그 중심 축을 따라 일정함을 말한다. 직교 프리미티브들 간의 교차 형상은 Fig. 1에서 보는 바와 같이 2차원 공간상에서 충분히 표현 가능하므로, 본 논문의 알고리즘을 설명하는 모든 예제 그림과 시스템의 구현에 2D 모델을 사용하였다.

## 2. 피쳐 인식

### 2.1 기존 연구

Graph pattern matching 방법은 Purdue대학의 T. Chang과 S. Joshi<sup>16)</sup>에 의해 처음 제안된 이래로 많은 연구가 되어왔다. 이는 솔리드 모델 면들의 위상정보(topology)를 이용하여 피쳐를 인식하는 방법이다. 모델내의 모든 면들을 노드(node)로 하고, 그들이 인접하고 있는 다른 면들과의 연결관계를 볼록(convex),

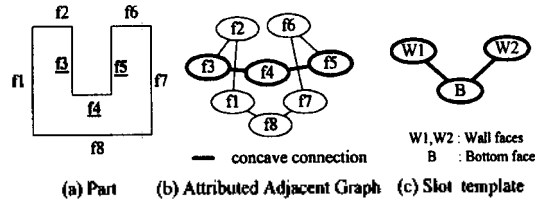


Fig. 2. Graph pattern matching.

오목(concave) 등의 속성(attributes)을 갖는 아크(arc)로 표현한 Attributed Adjacency Graph(AAG)를 이용하여 기본적인 피쳐의 패턴을 추출하는 방법이다(Fig. 2). 그런데 피쳐 간의 상호 교차가 발생하여 모델 내에 존재하는 피쳐 패턴의 일부 혹은 전부가 소실되는 경우, 미리 정의하여 둔 피쳐의 패턴이 더 이상 의미가 없어지는 단점이 있다.

가공을 시작해야할 최초 원 부재(stock)와 주어진 모델을 불리언 차집합하여 전체 가공해야할 볼륨을 계산 할 수 있는데, 이러한 볼륨을 델타볼륨(delta volume)이라 한다. Cell Decomposition & Composition 방법은 이러한 델타 볼륨내에 존재 하는 여러 면들 중 필요한 면들을 확장하여 반 공간(Half Space)을 생성하고, 이를 이용하여 델타 볼륨을 다수의 셀(Cell)들로 나눈다(Decomposing)(Fig. 3). 나누어진 셀들을 공구의 접근 방향이나 가공성등을 고려하여 다시 병합(Composing)함으로써 가공에 필요한 피쳐들을 인식하는 다단계 추론 방법이다<sup>17,18)</sup>. 이 방법의 문제점은 많은 불리언 연산수와 다단계 추론에 있는데, 예를 들면 첫 단계인 Cell Decomposition은 특정한 피쳐를 인식하겠다는 의도하에 수행되는 것이 아니라, 피쳐 인식이라는 목표로부터 완전히 분리되어 독자적으로 진행되며, 분해된 셀들을 병합하는데 있어서도 이제까지 제안된 알고리즘은 모두 임의의(ad hoc) 고안된 것이어서, 우리가 원하는 피쳐들을 항상 인식해 내지는 못하는 한계를 가진다<sup>12)</sup>.

Convex Hull Decomposition은 ASV(Alternating Sum of Volumes)<sup>14)</sup>을 기본으로 한 연구로 우선 대상 부품의 외부를 둘러싸는 볼록 겹질(Convex Hull)을

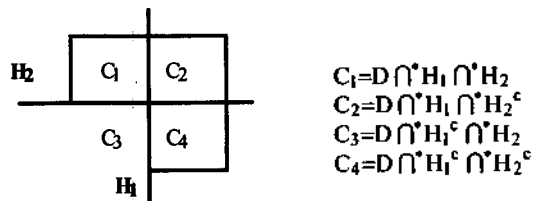


Fig. 3. Cell decomposition and composition.

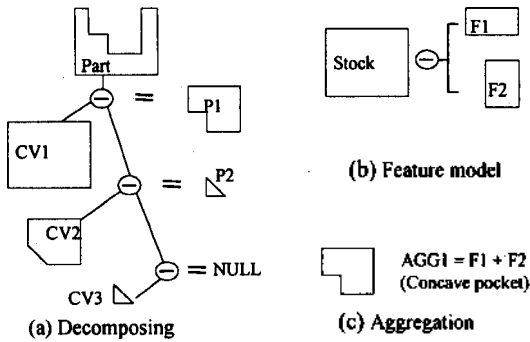


Fig. 4. Convex hull decomposition.

구한 후, 이 블록 집질의 볼륨에서 부품을 뺀다. 그 결과 얻어진 새로운 볼륨에 대해서 다시 블록 집질을 구하여 차 집합하는 방법을 NULL이 될 때 까지 반복하여 계산한다(Fig. 4(a)). 그 후 각 단계에서 계산된 볼륨 내 면들의 연결관계를 판단하고, 그 관계를 수정함으로써 최종적으로 피처를 인식한다(Fig. 4(b)). 그리고 ASV의 한계를 극복하기 위한 방법으로 체적을 분리하여 ASV를 수행하는 ASVP(Alternating Sum of Volumes with Partitioning) 방법이 Kim<sup>[9]</sup>에 의해 개발되었다. 특히 Waco와 Kim<sup>[10]</sup>은 인식된 피처들 간의 공간관계를 이용하여 서로 교차하고 있는 피처들 중에서 동일한 바다 면을 가지며, 동일한 공구 접근 방향을 갖는 피처를 결합하여 한번의 가공으로 제거될 수 있는 오목한 포켓을 찾도록 하였는데, 이를 피처 Aggregation이라 하였다(Fig. 4(c)). 그러나 이 방법 역시 셀 분해 방법과 마찬가지로 다단계 추론 과정을 거치며 유사한 한계를 가진다.

2.2 피처 인식의 문제점과 연구 방향

피처 인식 알고리즘들에 있어서 일반적인 문제점은 대부분의 경우, 피처들 간에 교차가 발생하는 경우에 인식이 어렵다는 것이다. 이는 피처 간의 교차로 인해 부품의 형상이 복잡해 지며, 교차하는 피처의 일부 또는 전체의 패턴이 깨지기 때문이다. 또 다른 문제점으로는, 피처 인식 과정이 가공 순서를 미

리 결정한다는 것이다. 다시 말해, 주어진 부품을 가공할 수 있는 여러 가지 가공순서 중 하나의 순서만을 인식 단계에서 미리 고려하여 피처를 인식함으로써, 다양한 가공순서를 만족 시키지 못하게 된다는 것이다. 이러한 문제를 극복하는 방법으로, 이미 인식된 피처들 간의 공간 관계나, 혹은 가공 관점의 경험 지식 등을 이용하여 또 다른 가공방법을 만족시킬 수 있는 피처를 찾아 주는 Alternative Feature Recognition이나 Feature Multiple Interpretation 등의 연구들이 수행되고 있다<sup>[11,12]</sup>. 그러나 이러한 방법들은 한번 인식한 피처를 다시 다른 형태로 인식해야 하는 번거로움이 있다. 이에 반하여 Sakurai는 Fig. 5의 (a)와 같이, 가공 순서에 무관하며 한번의 가공 작업(machining operation)으로 제거할 수 있는 최대한의 볼륨인 Maximal Feature Volume(최대 피처 볼륨)을 Cell Decomposition/Composition 방법으로 인식하고, 인식단계 이후에 가공 순서를 결정하는 방법을 제시하였다<sup>[7,8]</sup>. 또한 Vandenberg는 인식단계 이후에 Fig. 5의 (b)와 같이 하나의 피처 볼륨에 대하여 다른 피처들과 공간적 독립성을 갖는 부분(Required Volume: 필수 볼륨)과 그렇지 못한 부분(Optional Volume: 선택 볼륨)으로 나누어 결정된 가공 순서에 따라 실제 가공할 피처의 볼륨을 Fig. 5의 (c), (d)와 같이 계산하는 방법을 제시하였다<sup>[13]</sup>. 이를 통해 이미 인식된 피처를 이용하여 결정된 가공 순서에 적합한 가공 피처 모델을 생성하였다.

이에 본 연구에서는 피처가 상호 교차하는 경우에도 적합한 피처를 인식 할 수 있도록 하기 위해 피처들 간의 공간 관계에서 하나의 공간을 동시에 점유하고 있는 형상(교차 볼륨 또는 접촉면에 대한 정보를 추출하여 이들간의 공간 관계(상대 위치 관계)를 파악하여 피처를 인식하였다. 우선 부품의 모델링 시 사용된 프리미티브들로부터 후보 피처를 계산하고 이들 간의 공간관계를 표현하는 교차 형상 행렬을 구성하여 최대 피처 볼륨을 찾아 줌으로써, 설계 의도를 내포하고 있는 프리미티브로부터 가공에 필요한 피처 볼륨을 추적하는 보다 경제적인 방법을

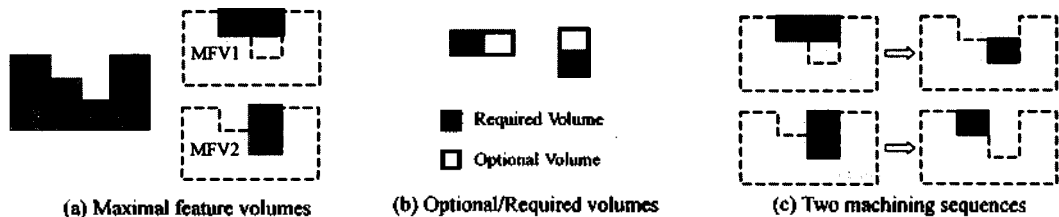


Fig. 5. Maximal feature volume and optional/required volumes.

취하였다. 그리고 인식 과정 중에 피쳐들 간의 교차 형상 정보를 교차 형상 행렬 내에서 계속적으로 갱신하여 좁으로써 가공할 피쳐의 선택 볼륨과 필수 볼륨을 계산하는 과정을 최대 피쳐 볼륨을 찾는 과정 내에 포함시켜 일원화 하였다.

### 3. 후보 피쳐 도메인 생성

#### 3.1 후보 피쳐 계산

솔리드 모델의 프리미티브는 설계자가 부품의 최종 형상과 기능을 고려하여 사용하는 것이므로 이는 곧 피쳐의 후보가 될 수 있다. 그러나, 일반적으로 Fig. 6의 (a)와 같이 솔리드 모델링 시에 불리언 연산에 참여하는 각 프리미티브는 실제 가공에 필요한 볼륨의 크기보다 더 크게 모델링 되는 경우가 많다. 그러므로 후보 피쳐는 원하는 부품을 생성하기 위해 가공을 시작할 볼륨인 원 부재(Stock)와 솔리드 모델 내의 각 프리미티브들 간의 교집합이 되어야 함을 알 수 있다<sup>[13]</sup>. 그러므로 본 연구에서는 식 (1)과 같이 원 부재와 솔리드 모델내의 프리미티브들 간의 교집합 연산으로 최초 후보 피쳐 도메인을 구성하였다.

$$CF = S \cap \bigcap P_i \tag{1}$$

where, S: Stock

$P_i$ : Design Primitives

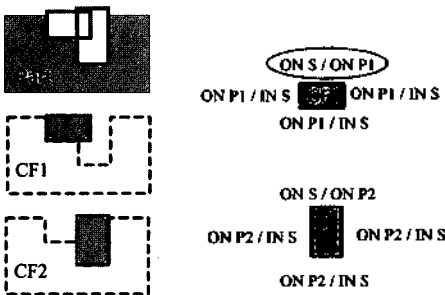


Fig. 6. Candidate feature calculation with face naming.

CF: Candidate features

$\cap$  \*\*: Regularized Boolean operation with Face Naming

이때 모델링 과정의 불리언 연산 시 사용자에게 의해 각 프리미티브들에 부여된 극성(포지티브 혹은 네거티브)을 후보 피쳐의 극성으로 상속하였으며, 또한 불리언 연산을 수행할 때 식 (1)에서 연산 기호가 뜻하는 바와 같이 불리언 연산(기준에 잘 알려진) 결과 이외에, 불리언 연산의 중간 단계인 set membership classification<sup>[14]</sup> 결과를 이용하여 Fig. 6의 (b)에서 보는 바와 같이 후보 피쳐의 각 면들에 정보를 갖는 이름(Name:속성)을 부여하였다. 구체적으로 각 후보 피쳐의 면들이 어떤 프리미티브들로부터 계산되었는지에 해당되는 상속 정보와, 각 면들이 부모(Parent) 프리미티브 내에서 어떤 위치에 존재하고 있었는지에 해당되는 상대 위치 정보를 부여 하였으며, 이를 본 연구에서는 Face Naming이라 명명하였다. 이렇게 이름이 붙은 면(named faces)들은 이후에 설명할 피쳐의 분할과 확장에 필요한 정보를 제공하며, 또한 공구의 접근 방향 검토를 통한 피쳐 분류에도 유용하게 사용된다.

#### 3.2 교차 형상 행렬

계산된 후보 피쳐들 간의 교차 유무와 그들의 공간적 관계를 규명하기 위해 각 후보 피쳐들 간의 교차 형상 정보가 필요하다. 후보 피쳐 간의 교차 형상의 계산 역시 3.1절의 식 (1)에서와 같은 방법으로 불리언 교집합 연산과 Face Naming을 동시에 수행한다. 그런데 후보 피쳐 간의 교차 유형은 Fig. 7에서 보듯이 두 피쳐가 서로 교차 볼륨을 가지는 볼륨교차(Volume Intersection)와 서로의 경계가 접촉하고 있는 경계교차(Boundary Intersection)의 두 가지 형태가 있다. 여기서 경계교차인 경우는 볼륨교차와는 달리 단순히 불리언 교집합 연산만을 이용하여 찾을 수 없으나, 이 역시 Face Naming의 결과를 이용하면 경계(접촉)면에서 set membership classification 결과

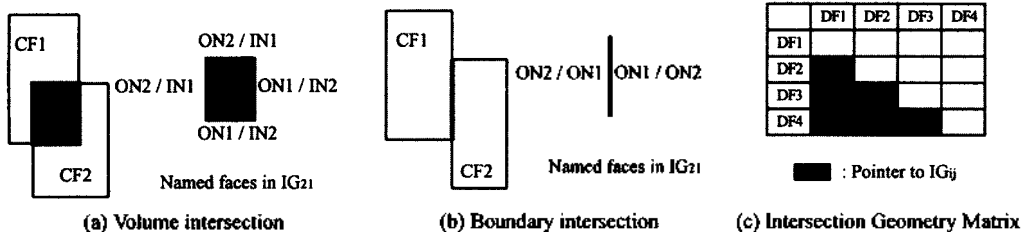


Fig. 7. Intersection geometry (IG) calculation with face naming.

가 ON i/ON j관계에 착안하여 경계교차인 경우에도 교차 형상(접촉면)을 찾을 수 있도록 하였다. 이렇게 계산된 후보 피쳐 간의 교차 형상들을 교차 형상 행렬(Intersection Geometry Matrix(IG<sub>ij</sub>)) 내에 저장하여 필요 시에 검색과 갱신을 용이하게 하였다. 교차 형상 행렬은 행과 열이 동일함으로 lower triangular matrix의 형태로 전체 후보 피쳐들 간의 교차 형상 정보를 모두 표현한다(Fig. 7(c)). 예를 들어 i번째 후보 피쳐와 다른 모든 후보 피쳐들과의 교차 형상을 검색하는 경우 교차 형상 행렬 내에서 IG<sub>ij</sub>(j=1~i-1)와 IG<sub>ij</sub>(j=i+1~n)를 검색하게 되는데, 특히 본 연구에서는 이들을 각각 후보 피쳐 i의 전 검색영역과 후 검색영역이라 하고, 그 둘을 합하여 피쳐 i의 전체 검색영역이라고 정의 하였다.

#### 4. 최대 피쳐 볼륨 계산

교차 형상 행렬을 이용하여 이미 계산한 후보 피쳐들 간의 공간 관계를 분석하여 필요에 따라 특정 후보 피쳐를 분할하거나 확장하여, 모든 후보 피쳐들이 2.2절에서 설명한 최대 피쳐 볼륨의 형태가 되도록 반 공간(half space) 개념을 이용한 후보 피쳐 분할과 확장 알고리즘을 개발하였다.

##### 4.1 후보 피쳐 분할(Candidate Feature dividing)

솔리드 모델의 프리미티브들 중에서 포지티브 극성을 갖는 프리미티브는 3.1절에 설명한 대로 포지티브 후보 피쳐를 생성한다. 그런데 이 포지티브 후보 피쳐가 네거티브 후보 피쳐와 불륨교차를 하는 경우, 이것은 가공되어서는 안 되는 볼륨을 의미한다. 따라서 포지티브 후보 피쳐와 교차하고 있는 네거티브 후보 피쳐의 볼륨을 가공 가능한 여러 개의 볼륨으로 나누어야 한다. 포지티브 후보 피쳐와 교차하고 있는

네거티브 후보 피쳐를 다음과 같이 네 단계를 통하여 가공 가능한 볼륨으로 분할할 수 있다.

##### 4.1.1 후보 피쳐 분할 알고리즘

단계 1: 하나의 포지티브 후보 피쳐의 전 검색영역에서 불륨교차 하는 네거티브 후보 피쳐를 찾는다.

단계 2: 아진 교차 볼륨 내에서 포지티브 후보 피쳐로부터 상속 받은 면을 이용하여 그 면의 법선 방향으로 반공간을 생성한다.

단계 3: 생성된 반공간을 단계 1에서 찾은 네거티브 후보 피쳐와 교차 시켜 새로운 네거티브 후보 피쳐를 생성한다.

단계 4: 필요한 교차 형상을 계산하여 단계 3에서 계산된 새로운 후보 피쳐와 함께 후보 피쳐 도메인을 갱신한다.

위 알고리즘을 Fig. 8을 예로 들어 설명하면, 우선 (a)와 같이 네거티브 후보 피쳐인 CF1(-)과 포지티브 후보 피쳐인 CF2(+)가 상호 불륨교차를 하는 경우, 단계 1에서는 교차 형상 행렬을 이용하여 CF2(+)의 검색영역 내에서 불륨교차를 하는 CF1(-)을 찾는다. 그러나 이 단계에서 CF2(+)의 전체 검색영역을 다 검색하지는 않으며 단지 전 검색영역만을 검색하게 된다. 그 이유는 만약 하나의 포지티브 후보 피쳐의 후 검색영역 내에서 다른 네거티브 후보 피쳐와 불륨교차를 한다고 해도, 찾아진 네거티브 후보 피쳐는 모델링 시의 연산 순서상, 현재 고려하고 있는 포지티브 후보 피쳐 보다 이후에 연산 된 것이므로 그 교차 볼륨은 가공해도 무방한 볼륨이 되기 때문이다. 그러므로 가공 해서는 안 되는 교차 볼륨은 항상 대상 포지티브 후보 피쳐의 전 검색영역에만 존재한다. 다음으로 단계 2에서는 Fig. 8의 (b)와 같이 CF2(+)와 CF1(-)의 교차 볼륨 내에 존재하는 면들 중에서 CF2(+)로부터 상속 받은 두개의 면을 찾고, 그 면

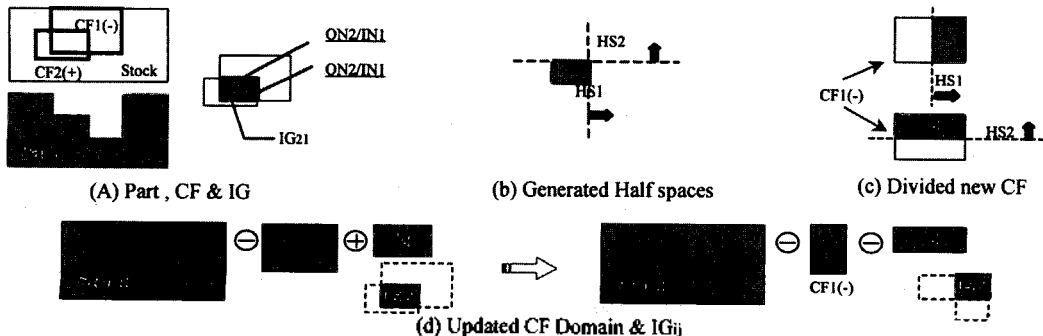


Fig. 8. Candidate feature dividing algorithm.

의 법선 방향으로 각각 반 공간 HS1과 HS2를 생성한다. 단계 3에서는 Fig. 8의 (c)와 같이 CF2(+)와 교차 하고 있는 네거티브 후보 피쳐인 CF1(-)을 단계 2에서 생성한 두개의 반공간과 차례로 교차 시켜 두개의 새로운 네거티브 후보 피쳐를 생성한다. 마지막 단계 4에서는 단계 3에서 새롭게 계산된 두개의 후보 피쳐와 이들 간의 교차 형상을 계산하여 전체 후보 피쳐 도메인을 Fig. 8의 (d)와 같이 갱신한다.

이상과 같은 피쳐 분할 알고리즘을 모든 포지티브 후보 피쳐에 적용하면 Fig. 8의 (d)에서 보듯이 전체 후보 피쳐 도메인 내에는 가공에 의미 있는 네거티브 후보 피쳐들만이 존재 하게 된다. 그러나 이러한 네거티브 후보 피쳐들은 아직도 최대 피쳐 볼륨의 형태는 아니므로 후보 피쳐의 확장 알고리즘을 통해 최대 피쳐 볼륨을 계산 하였다.

**4.2 후보 피쳐 확장(Candidate Feature growing)**

후보 피쳐 도메인 내에 피쳐들 간의 교차 형상에 대한 정보를 교차형상 행렬 내에 이미 가지고 있으므로 이를 이용하여, 4.1절의 후보 피쳐의 분할 알고리즘과 유사한 피쳐 확장 알고리즘을 개발하였다. 구체적인 알고리즘은 다음과 같다.

**4.2.1 후보 피쳐 확장 알고리즘**

- 단계 1: 하나의 네거티브 후보 피쳐의 전체 검색영역에서 교차하는 네거티브 후보 피쳐를 찾는다.
- 단계 2: 이들의 교차형상 내에서 확장성을 검토하고자 하는 피쳐로부터 상속 받은 면 중에서 자신의 원래 면적과 동일한 면을 찾아 그 면의 법선 방향으로 경계를 가지는 반공간을 생성한다.
- 단계 3: 생성된 반공간을 상대 네거티브 후보 피쳐와 교차 시켜 확장할 볼륨을 계산하고 이를 더하여 피쳐를 확장한다.
- 단계 4: 필요한 교차 형상을 계산하여 단계 3에서

계산된 새로운 후보 피쳐와 함께 후보 피쳐 도메인을 갱신한다.

위와 같은 알고리즘을 역시 Fig. 9를 예로 들어 설명하면 다음과 같다. 여기서 확장을 검토할 피쳐로 CF1(-)을 기준으로 설정한 경우를 '갑'이라 하고, 반대로 CF2(-)을 기준으로 설정한 경우를 '을'이라 하여 설명한다. Fig. 9의 (a)와 같이 CF1(-)이 CF2(-)와 볼륨교차를 하는 경우, 단계 1에서는 갑의 경우에, 교차 형상 행렬 내 CF1(-)의 전체 검색영역에서 교차하고 있는 CF2(-)를 찾고, 을의 경우도 마찬가지로 CF1(-)을 찾는다. 단계 2에서는 갑의 경우에, 이들 간의 교차 볼륨 내에 존재하는 면들 중에서 Fig. 9의 (a)와 같이 CF1(-)으로부터 상속 받은 두개의 면을 찾고, 찾아진 두개의 면 중에서 점선으로 표시된 면과 같이 원래 자신의 면적과 동일한 면적을 갖는 확장면을 선택하고 Fig. 9의 (b)와 같이 그 면적을 기준으로 하여, 면의 법선 방향으로 경계를 가지는 반공간(bounded half space; B\_HS)을 생성한다. 그러나 을의 경우에는 교차 볼륨 내에서 찾아진 CF2(-)으로부터 상속 받은 두개의 면이 모두 원래 자신의 면적과 동일한 면적을 갖지 않으므로 검사를 중단하고 루프를 탈출한다. 그러므로 갑의 경우만 단계 3, 4를 계속 수행한다. 단계 3에서는 단계 2에서 생성한 반공간을 CF1(-)이 교차하고 있는 CF2(-)와 교차 시켜 Fig. 9의 (c)와 같이 확장할 볼륨을 생성하고 이를 CF1(-)에 더하여 새로운 네거티브 후보 피쳐를 생성한다(Fig. 9(d)). 마지막으로 단계 4에서는 4.1절의 후보 피쳐 분할 알고리즘과 마찬가지로 필요한 교차 볼륨을 계산하여 단계3에서 계산된 확장된 피쳐와 함께 후보 피쳐 도메인을 갱신한다(Fig. 9(e)). 그리고 Fig. 10은 두개의 네거티브 피쳐가 경계 교차를 하는 경우에 대해 확장 알고리즘을 적용한 예를 보여 주고있다. 이 경우에도 앞서 설명한 방법과 동일하게 확장된 피쳐를 효과적으로 얻을 수 있다.

이상과 같은 피쳐 확장 알고리즘을 모든 네거티브

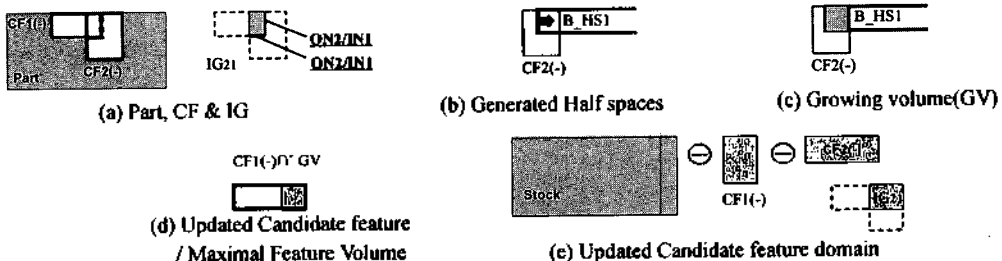


Fig. 9. Candidate feature growing algorithm.

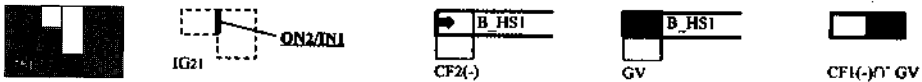


Fig. 10. Growing algorithm in the case of boundary intersection.

후보 피쳐들에 대해 적용하여 후보 피쳐 도메인 내의 모든 네거티브 피쳐들이 최대 피쳐 볼륨을 갖도록 한다. 또한 알고리즘을 수행하는 동안 계속하여 갱신되어 온 교차 형상 행렬 내에는 최대 피쳐 볼륨들 간의 교차 형상들이 존재한다. 이후에 가공 경험지식을 적용하여 가공 순서가 결정된 후 그 순서에 알맞은 실제 가공 피쳐 모델을 생성하는데 있어서, 교차 형상 행렬은 앞서 2.2절에 설명한 필수/선택 볼륨을 선정할 때 중요한 정보를 제공한다.

### 5. 사례 연구

본 연구에서 개발된 알고리즘을 검증하기 위하여 2D에 적용 가능한 시스템을 구현하였다. 본 시스템의 하드웨어로는 IBM RS/6000 355를 사용하였으며, 개발 언어로는 솔리드 모델의 자료구조를 표현하기에 적절한 C 언어를 사용하였다. 화면 디스플레이를 위해서는 GL Library를 사용하였고 시스템의 전체 구조는 Fig. 11과 같다.

#### 5.1 가공 피쳐 모델 생성

본 연구에서는 기존의 연구자들<sup>10)</sup>에 의해 사용된 몇 가지 통상적인 경험지식을 적용하여 가공 순서를 결정 하도록 하였다. 첫째로 가장 많은 피쳐를 포함하는 공구의 접근 방향(setup)을 먼저 가공하게 함으로써 전체 setup 수를 최소화 하였다. 둘째로 하나의 setup 내에서 가장 큰 공구로부터 가공 하게 함으로써 공구의 접근성을 양호하게 하였다. 그리고 셋째로 한번에 가공 가능한 최대한의 볼륨을 가공 하게 함으로써 전체 가공 횟수를 줄일 수 있도록 하였다.

결정된 가공 순서에 따른 가공 피쳐 모델을 생성하기 위하여 하나의 최대 피쳐 볼륨을 다른 피쳐와

무관한 부분인 필수 볼륨(Required Volume)과, 그렇지 못한 부분인 선택 볼륨(Optional Volume)으로 나누어 가공 피쳐를 계산 하였다.

#### 5.2 필수/선택 볼륨으로 가공 피쳐 계산

피쳐 분할과 확장을 통해 계산된 최대 피쳐 볼륨들 중에 각각의 가공 작업에서 서로 중복될 가능성이 있으며, 그들의 교차 볼륨이 다음과 같은 조건을 만족하는 경우에만 이 교차 볼륨이 두 개의 피쳐 상호 간의 선택 볼륨이 된다.

조건 1: 교차 볼륨 내의 면들이 연속적으로 ON  $i$ /IN  $j$ 속성을 갖지 않아야 한다.

조건 2: 교차 볼륨 내의 면들이 서로 마주보며 ON  $i$ /IN  $j$  속성을 갖지 않아야 한다.

첫번째로 조건 1은 Fig. 12의 (a)와 같은 경우로서, 두개의 피쳐 볼륨이 서로 교차 하고 있으나 이 볼륨은 하나의 피쳐가 먼저 가공된다 할 지라도 다음 피쳐를 가공할 때 공구가 반드시 그 교차 볼륨을 지나 가야 하는 경우이다. 두 번째로 조건 2는 두개의 피쳐가 서로 Fig. 12의 (b)와 같이 서로 가로지르는 경우로서, 이 교차볼륨을 선택 볼륨으로 선정하게 되면 하나의 가공 이후에 나머지 피쳐는 두개의 볼륨으로 나누어져 가공되어야 하므로 전체 가공 횟수가 늘어 나게 된다. 그렇게 되면 앞서 설정한 가공 순서 결정의 세 번째 기준에 위배된다.

후보 피쳐 도메인 내에서 계산된 최대 피쳐 볼륨에 대하여 이전 단계에서 결정된 필수 볼륨과 선택 볼륨을 이용하여 가공 순서에 따라 가공 되어야 할 피쳐의 볼륨을 계산하였다. 만약 어떤 피쳐가 가공 순서상으로 먼저 가공될 피쳐와 선택 볼륨을 가지는 경우, 원래 피쳐의 볼륨에서 그 선택 볼륨을 차집합 연산한 결과는 실제 가공 피쳐가 된다. 예를 들면 2.2절에서 예로 든 Fig. 5의 (a), (b)와 같이 최대 피쳐

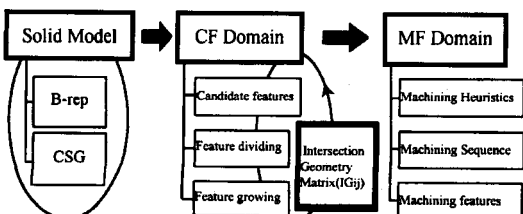


Fig. 11. System structure.

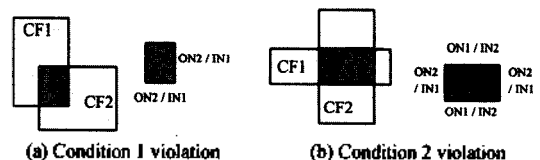


Fig. 12. Optional volume conditions.

블룸과 선택 블룸을 갖는 경우에 가공 순서는 MFV 2 → MFV1 이고 최종 가공 피쳐 모델은 Fig. 5의 (d)와 같이 두 개의 슬롯이 된다.

5.3 적용 사례

본 시스템에 대한 두 가지 특징적인 사례를 살펴보기로 한다. 먼저 Fig. 13의 사례는 (a)와 같은 부품을 (b)와 같이 서로 복잡하게 교차하고 있는 프리미티브들을 이용하여 구성하였을 때, 본 연구에서 개발한 피쳐 인식 알고리즘을 적용한 예로서, (c)는 주어진 부품에 대하여 피쳐의 분할과 확장을 통해 최대 피쳐 블룸을 찾아준 모습이며, (d)는 결정된 가공 순서에 따라 계산된 다섯 개의 슬롯(slot)을 순차적으로 보여 주는 장면이다. 그리고 Fig. 14은 여섯 개의 슬롯과 두 개의 스텝(step)이 존재하는 모델에 대하여 시스템이 인식된 피쳐들을 결정된 가공 순서에 따라 순차적으로 보여주는 장면으로, 특히 (e)와 (g)는 두 개의 슬롯이 서로 교차하여 상호간의 교차블룸(그림(e), (g)에서 동그라미 표시 부분)을 가지고 있

으나, 5.2절에 설명한 조건 1(Fig. 12(a))을 위반하여 상호간의 선택 블룸이 될 수 없는 경우이다. 이 교차 블룸은 그림과 같이 두개의 가공 작업에 모두 포함된다.

6. 결 론

본 연구에서는 솔리드 모델 내의 프리미티브들을 이용하여 후보 피쳐 도메인을 생성하고, 이들의 공간적 관계를 표현하기 위한 교차형상 행렬을 구성한 후 피쳐의 분할과 확장을 통해 가공 피쳐를 인식하는 시스템을 개발 하였다.

본 시스템은 설계 프리미티브간의 교차가 있어 복잡한 경우에도 시스템이 내부에 프리미티브 간의 교차형상 정보를 가지고 있으므로 필요한 가공 피쳐를 인식할 수 있으며, 또한 인식 단계에서 가공 순서에 무관한 피쳐 블룸(최대 피쳐 블룸)을 생성하므로 가공 순서 결정 기준의 변화에 용이하게 대처 할 수 있다. 그 외에도 인식 작업이 끝난 서로 다른 부품에 대하여 그들 각각의 교차 형상 정보들을 검사하여 두 부품간의 간섭영역을 확인 하는 등의 여러 응용 범위에 효과적으로 이용할 수 있을 것이다.

그러나 한계점으로는, 본 알고리즘을 Fig. 15와 같이 직교 교차 형상이 아닌 일반적인 경우에 직접 적용할 수 없다는 것이다. 구체적으로 설명하면, 블룸 자체를 분할하거나 확장하는 데에는 문제가 없지만 결과 블룸의 형상이 매우 복잡하게 되어, 그들에 대한 가공 방법을 각각 달리 하여야 한다. 그러므로 일정한 기준(예를 들어, 프리미티브 중심 축 간의 교차 각도)에 따라 가공 방법을 달리 한다거나, 아니면 새로운 피쳐를 정의 하는 일반적인 방법에 관한 연구 등이 더 필요하다고 하겠다.

아울러, 본 시스템과 관련하여서는 직교가 아닌 일반적인 교차 형상을 갖는 경우에도 적용되는 알고리즘에 대한 계속적인 연구가 필요하며, 3차원 모델에 적용 후, 주어진 CAD 모델에서 가공 가능한 모든 최대 피쳐 블룸을 찾는 기존 연구 방법들<sup>8,12</sup>과의 계

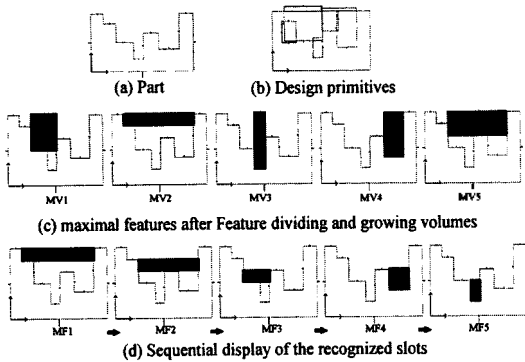


Fig. 13. Example 1 (color edited).

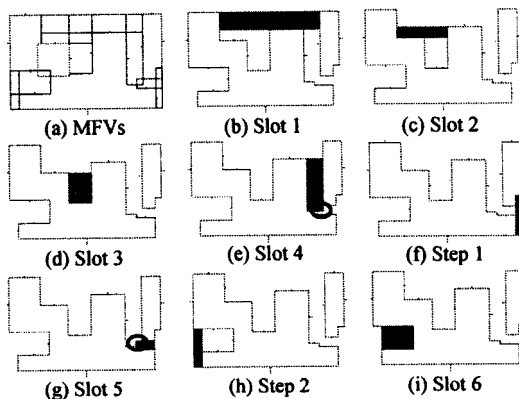


Fig. 14. Example 2.

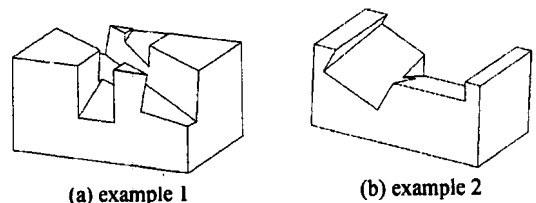


Fig. 15. Complex intersection geometry.



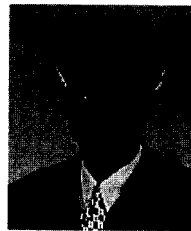
산과정 상의 비교우위를 검토하는 것 또한 흥미로운 과제라 생각된다.

**참고문헌**

1. Shah, J. and Mantyla, M., *Parametric and Feature-based CAD/CAM*, A Wiley-Interscience Publication, Jhon Wiley & Sons, Inc., New York, 1995.
2. 한정현 "특징형상 인식 연구의 현주소" 한국 CAD/CAM 학회지 제 3권, 제 2호, pp. 66-69, 1997.
3. Shah, J. Mantyla, M. and Nau, D. editors, *Advances in Feature Based Manufacturing*, Elsevier Science B. V., Amsterdam, The Netherlands, 1994.
4. Woo, T., "Feature Extraction by Volume Decomposition", *Proc. Conf. CAD/CAM Tech. In Mechanical Engineering*. Cambridge, MA, Mar, 1982.
5. Kyprinanou, L.K., "Shape Classification in Computer Aided Design", Ph.D. thesis, Stanford University, 1980.
6. Joshi, S. and Chang, T., "Graph based Heuristics for Recognition of Machined Features From 3-D Solid Model", *CAD*, Vol. 20, pp. 58-66, 1988.
7. Sakurai, H. and Chin, C., "Definition and Recognition of Volume Features for Process Planning", In.: Shah, J.J. Mantyla, M. and Nau, D.S. editors, *Advances in Feature Based Manufacturing*, pp. 65-80. Elsevier Science B.V., Amsterdam, The Netherlands, 1944.
8. Dave, P. and Sakurai, H., "Maximal Volume Decomposition and its Application to Features Recognition", *In proc. ASME Computer in Engineering Conf.*, pp. 553-568, 1995.
9. Kim, Y., "Recognition of Form Features Using Convex Decomposition", *CAD*, Vol. 24, No. 9, pp. 461-476, 1992.
10. Waco, D. and Kim, Y., "Considerations in Positive to Negative Conversion for Machining Features using Convex Decomposition", *In Proc. ASME Computers in Engineering Conf.*, pp. 35-45, 1993.
11. Gupta, S.K., "Automated Manufacturability Analysis of Machined Parts", Ph.D. dissertation, Mech. Eng. Dept., Univ. of Maryland. 1995.
12. Tseng, Y. and Joshi, S., "Recognizing Multiple Interpretations of Interacting Machining Features",

*CAD*, Vol. 26, No. 9, pp. 669-688, 1994.

13. Vandenbrande, J. and Requicha, A., "Geometric Computation for the Recognition of Spatially Interacting Machinable Features", In.: Shah, J. Mantyla, M. and Nau, D., editors, *Advances in Feature Based Manufacturing*, Elsevier Science B.V., Amsterdam, The Neterland, 1994.
14. Requicha, A.A.G. and Voelcker, H.B., "Boolean Operation in Solid Modeling: Boundary Evaluation and Merging Algorithm", *Proc. IEEE in Mechanical Engineering*, Oct., pp. 30-44, 1983.
15. Kambawi, H. Leclair, S. and Chen, C., "Feature sequencing in the rapid design system using a genetic algorithm", *Journal of Intelligent Manufacturing*, Vol. 7, No. 1, pp. 55-67, 1996.
16. Sormaz, D. and Khoshnevis, B., "Process sequencing and process clustering in process planning using state space search", *Journal of Intelligent Manufacturing*, Vol. 7, No. 3, pp. 189-200, 1996.



**정 채 범**

1996년 한양대학교 기계공학과 학사  
 1998년 한양대학교 기계설계학과 석사  
 1998년 ~ 현재 한양대학교 기계설계학과 박사과정  
 관심분야 : Geometric Modeling, Feature Recognition, Parametric Modeling



**김 재 정**

1981년 한양대학교 경일공학과 학사  
 1983년 미국 George Washington 대학 공학석사  
 1983년 ~ 1984년 미국 National Food Processors Association 연구원  
 1989년 미국 MIT 공학박사  
 1989년 ~ 1991년 미국 IBM T.J. Watson 연구소 연구원  
 1991년 ~ 1993년 한국 IBM 소프트웨어 연구소 연구원  
 1993년 ~ 현재 한양대학교 기계공학부 부교수  
 관심분야 : Geometric Modeling, CAD/CAM, 응용