

&lt;응용논문&gt;

DOI <http://dx.doi.org/10.3795/KSME-A.2016.40.3.327>ISSN 1226-4873(Print)  
2288-5226(Online)

## 형상 검색을 이용한 제트엔진 절삭가공을 위한 빠른 CAM 모델 생성 방법

김병철<sup>\*†</sup> · 송일환<sup>\*\*</sup> · 신수철<sup>\*\*</sup>

\* 동아대학교 기계공학과, \*\* 한화테크윈 CAE 기술그룹

### A Fast Generation Method of CAM Model for Machining of Jet Engines Using Shape Search

Byung Chul Kim<sup>\*†</sup>, Ilhwan Song<sup>\*\*</sup> and Suchul Shin<sup>\*\*</sup>

\* Dept. of Mechanical Engineering, Dong-A Univ.,

\*\* CAE Technology Group, Hanwha Techwin

(Received November 25, 2015 ; Revised December 18, 2015 ; Accepted December 22, 2015)

**Key Words:** Computer-Aided Manufacturing(컴퓨터 이용 제조), Computer-Aided Design(컴퓨터 이용 설계), Jet-Engine Compressor Case(제트엔진 압축기 케이스), Shape Search(형상 검색)

**초록:** 항공기 엔진 제작사들은 수치제어 공작기계를 운영하고 제어하기 위해 CAM 소프트웨어를 도입했다. 그러나 CAM 모델을 생성하는 일은 긴 시간이 걸리고 오류가 발생하기 쉽다. 이는 가공연산 및 절차를 수작업으로 정의하기 때문이다. CAM 모델을 자동으로 생성하기 위해, 특징형상 인식 기술들이 오래 전부터 연구되었다. 그러나 인식 범위가 제한적이기 때문에 제트엔진과 같이 복잡한 형상에는 완전히 적용할 수 없다. 본 연구에서는 형상 검색 기술을 이용해 CAD 모델로부터 CAM 모델을 빠르게 생성하는 새로운 방법을 제안한다. 이 방법에서는, 작업자가 한 개의 가공연산을 참조연산으로 지정하면, 동일한 가공형상을 가지는 형상들을 검색하고, 참조연산을 검색된 형상들의 위치로 복사한다. 제트엔진 압축기 케이스를 대상으로 한 실험을 통해 제안한 방법을 검증하였다.

**Abstract:** Manufacturers of aircraft engines have introduced computer-aided manufacturing (CAM) software to operate and control computerized numerical control (CNC) machine tools. However, the generation of a CAM model is a time consuming and error-prone task since machining procedure and operational details are manually defined. For the automatic generation of a CAM model, feature recognition techniques have been widely studied. However, their recognition coverage is limited so that complex shapes such as a jet engine cannot be fully developed. This study presents a novel approach to quickly generate a CAM model from a CAD model using shape search techniques. Once an operator sets a machining operation as a reference operation, the same shapes as the shapes referenced by the operation are searched. The reference operation is copied to the positions of the searched shapes. The proposed method was verified through experiments with a jet engine compressor case.

#### 1. 연구 배경

항공기 제트엔진은 높은 정밀도로 생산해야 하기 때문에 고수준의 가공 기술이 필요하며, 주로 절삭가공으로 생산한다. 그러나 사람이 수치제어

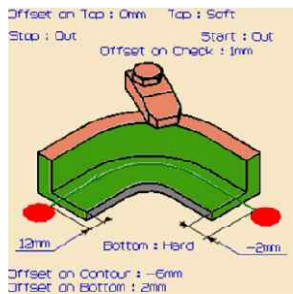
(computerized numerical control, CNC) 공작기계를 이용해 직접 가공하는 것은 가공 오차 및 오류를 발생시킬 수 있다. 따라서, 항공기 제트엔진 제작사들은 CAM(computer-aided manufacturing, 컴퓨터 이용 제조) 소프트웨어를 도입하여 활용한다. CAM 소프트웨어를 이용할 때에는, 사전에 설계된 CAD(computer-aided design, 컴퓨터 이용 설계) 모델을 이용해 작업자가 컴퓨터 상에서 가공 경로를 설정하여 CAM 모델을 생성한다. 또한 시뮬레이션을

† Corresponding Author, mir7942@dau.ac.kr

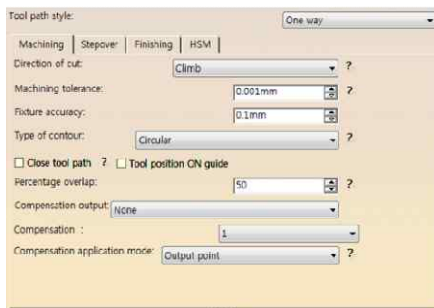
© 2016 The Korean Society of Mechanical Engineers



(a)



(b)

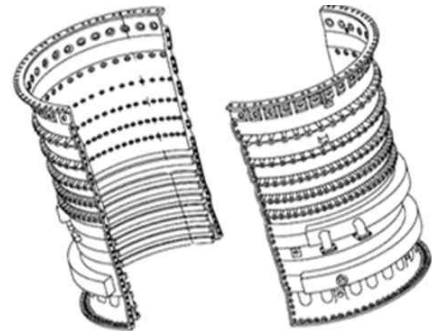


(c)

**Fig. 1** Example of a CAM model in CATIA V5: (a) machining operations, (b) setting reference shapes, and (c) machining strategies

통해 가공 시 발생할 수 있는 문제점을 사전에 확인한 후, CAM 모델로부터 G코드를 생성해 수치제어 동작기계를 이용한 절삭 가공에 들어간다.

CAM 모델은 CAD 모델을 기반으로 형상을 가공하는데 필요한 공구 경로 설정 정보, 절삭 조건, 가공전략 등을 포함한 모델이다. 가공경로를 명시적으로 포함하지 않고, 형상을 가공하는데 필요한 가공연산(machining operation) 또는 가공 특징형상(machining feature)을 이용해 표현한다. Fig. 1(a)는 CATIA에서 정의한 CAM 모델의 일부이다. Fig. 1(a)에 보면, 형상을 가공하기 위해 ‘Profile Contouring’, ‘Multi-Axis Curve Machining’ 등의 가공연산을 정의한 것을 볼 수 있다. 또한 각각의 가공연산을 정의하기 위해서는 Fig. 1(b)에서와 같이 참조형상을 정의해 주고, Fig. 1(c)에서와 같이 다양한 가공전략들을 정의해 준다.



**Fig. 2** The jet-engine compressor case image from the internet<sup>(1)</sup>

Fig. 1(b)에서와 같이, 가공연산은 CAD 모델에 있는 면(face)이나 모서리(edge)를 참조해서 정의한다. 이 과정에서는 사용자의 판단 하에 적절한 면과 모서리를 선택하는 것이 가장 중요하다. 따라서 가장 많은 주의가 필요하며, 가장 많은 시간이 걸리는 부분이기도 하다. 더군다나 마땅히 참조할 만한 면이나 모서리가 없는 경우에는 직접 면이나 모서리를 생성해 주어야 한다. Fig. 2는 제트엔진의 일부인 압축기 케이스를 보여준다. Fig. 2를 보면 형상이 매우 복잡한 것을 알 수 있다. 이처럼 복잡한 형상에 대해 수작업으로 가공연산을 정의할 때에는, 작업시간이 짧게는 수 일에서 길게는 수 주까지 걸린다.

이러한 문제 해결을 위해 가공연산을 자동으로 인식하기 위한 특징형상 인식 방법<sup>(2-4)</sup>이 1980년대부터 널리 연구되었다. 일부 특징형상 인식 방법들은 상용 소프트웨어에 구현되어 있다. 그러나 특징형상 인식 방법을 항공기 제트엔진과 같이 복잡하면서 형상에 오류가 존재하는 대상에 적용할 경우에는 인식률이 낮다. 또한 인식된 부분에 대해서도 실제 작업자가 원하는 것과 다른 결과를 제시하는 경우가 많다.

본 연구에서는 가공 특징형상 인식에 대한 대안으로 형상 검색을 이용한 방법을 제안한다. 이는 반자동으로 빠르게 CAM 모델을 생성할 수 있는 방법이다. 이 방법에서는 사용자가 정의한 가공연산을 다른 곳에 복사해 사용한다. 복사할 위치는 형상 검색을 통해 찾는다. 이 방법을 이용하면 매우 빠른 시간 안에 CAM 모델을 생성할 수 있다.

본 논문은 다음과 같이 구성된다. 2절에서는 관련 연구에 대해 살펴본다. 3절에서는 본 연구에서 제안하는 방법에 대해 설명한다. 4절에서는 구현 및 실험 결과를 제시하고, 5절에서 연구 내용 요약으로 본 논문을 마무리한다.

## 2. 관련 연구

CAD와 CAM을 통합시키기 위해 CAD 모델로부터 가공 특징형상 또는 가공연산을 자동으로 인식하기 위한 연구가 1980년대부터 진행되었다. 상당수의 연구가 진행되었으며, 대표적 연구내용 및 각 방법의 장·단점은 참고문헌 (2)와 (3)에 잘 정리되어 있다. 대표적인 특징형상 인식 방법으로는 그래프 기반(graph-based) 방법,<sup>(5-8)</sup> 볼륨 분해(volume decomposition) 방법,<sup>(9-15)</sup> 힌트 기반(hint-based) 방법<sup>(16,17)</sup>이 있다.

그래프 기반 방법에서는 원본 형상과 특징형상의 면과 모서리 관계를 그래프(graph)로 표현하여 그래프 검색을 통해 특징형상을 인식한다. 이 방법에서는 인식할 새로운 특징형상을 쉽게 추가할 수 있고, 다양한 도메인에 적용할 수 있다는 장점이 있다. 그러나, 인식할 수 있는 특징형상의 위상(topology) 구조가 가변적인 경우나 특징형상이 교차하는 경우에는 적용하기 어렵다는 것이 단점이다. 또한 그래프 검색에는 지수적인 시간이 걸리기 때문에 제트엔진과 같은 복잡한 모델에는 적용하기가 어렵다.

볼륨 분해 방법은 형상을 작은 볼륨들로 분해하고, 작은 볼륨들을 특징형상으로 인식하는 방법이다. 볼륨을 분해하는 방법에 따라 컨벡스 분해(convex decomposition)<sup>(9,10)</sup>와 셀 기반 분해(cell-based decomposition)<sup>(11-15)</sup>로 나뉜다. 컨벡스 분해는 원본 형상을 형상의 컨벡스셸(convex hull)과 델타 볼륨(delta volume)으로 순차적으로 분해해 나가는 방법이고, 셀 기반 분해는 원본 형상을 단순한 형상을 가진 셀(cell)로 분해하고, 이 셀들을 큰 볼륨으로 재조합 하는 방법이다. 볼륨 분해 방법은 특징형상이 교차하는 경우에도 적용하기 쉽고, 인식된 특징형상이 면이 아닌 볼륨(volume)으로 표현된다는 장점이 있다. 그러나 계산 시간이 오래 걸리기 때문에, 제트엔진과 같은 복잡한 모델에 대해 적용할 경우 합리적인 시간에 결과를 얻기가 어렵다.

힌트 기반 방법은 특징형상의 완전한 패턴을 찾는 대신 특징형상 인식을 위한 최소한의 흔적(trace)에서 시작하여, 주변 형상에 대해 기하추론 과정을 거쳐 특징형상을 찾는다. 그러나 특징형상이 다른 특징형상과 교차되는 경우에는 특징형상의 일부 형상이 사라지기 때문에 존재하는 형상만으로는 온전한 특징형상을 정의할 수 없다. 따라서 힌트 기반 방법에서는 흔적과 흔적 주변의 형

상을 이용해 특징형상을 복원(geometric completion)한다. 이는 특징형상 인식의 가장 큰 어려움인 교차 특징형상 문제를 해결해 준다. 힌트 기반 방법은 흔적으로부터 검색을 시작하기 때문에 인식 속도가 빠르다는 장점이 있다. 그러나 힌트 기반 방법은 각각의 특징형상에 대해 인식 규칙을 개별적으로 정의해 주어야 한다는 단점이 있다.

이와 같이 매우 많은 특징형상 인식 연구가 진행되었음에도 불구하고, 아직까지 일반적으로 적용할 수 있는 특징형상 인식 방법은 없다. 그리고 대부분의 방법들이 필렛이나 라운드 형상이 있는 경우 적용이 어렵기 때문에 사전에 이를 제거해야 사용할 수 있다. 그러나 복잡한 형상의 경우에는 필렛이나 라운드를 제거하는 것이 불가능한 경우가 많다. 또한 개발 업체 간에 CAD 파일을 교환하는 과정에서 형상에 오류가 생기는 경우가 빈번하며, 이러한 형상 오류는 인식을 방해한다. 또한 인식된 부분에 대해서도 실제 작업자가 원하는 것과 결과가 다른 경우가 많다. 따라서 본 연구에서는 특징형상 인식 방법에 대한 대안으로 형상을 검색해 가공연산을 복사하는 방법을 제안한다.

## 3. 형상 검색을 이용한 CAM 모델 생성 방법

### 3.1 전체 절차

이번 절에서는 본 연구에서 제안하는 형상 검색을 이용한 CAM 모델 생성 방법에 대해 설명한다. Fig. 3은 제안하는 방법의 절차를 보여준다.

첫 번째로, 작업자는 한 개의 가공연산을 참조연산(reference operation)으로 선택한다. 또한 형상 검색 시 참조하기 위한 한 개 이상의 참조형상(reference shape)을 선택한다. 예를 들면, 드릴링 연

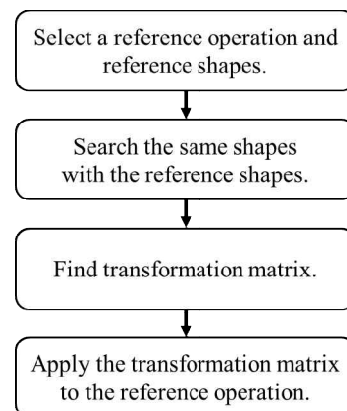


Fig. 3 Procedure of the proposed method

산(drilling operation)을 참조연산으로 선택할 경우, 구멍 가공면에 해당하는 원통 곡면(cylindrical surface)이나 구멍 입구에 해당하는 원형 곡선(circular curve)을 참조형상으로 선택할 수 있다. 참조형상을 선택할 경우에는 가공연산 정의 시 사용한 형상뿐만 아니라, 가공 형상 주변에 공통적으로 나타나는 임의의 형상을 선택할 수 있다.

두 번째로, 참조형상과 동일한 형상을 CAD 모델 내에서 검색한다. 이 때 검색은 형상 검색 방법<sup>(17)</sup>을 이용해 자동으로 진행된다. 본 연구<sup>(8)</sup>에서는 다양한 형상 검색 방법 중 전역 특징 기반 방법(global feature-based method)을 사용한다. 이에 대한 내용은 3.2절에서 설명한다. 세 번째로, 참조형상을 검색된 형상의 위치로 복사하기 위한 변환행렬을 계산한다. 이에 대한 내용은 3.3절에서 설명한다. 마지막으로 참조연산에 변환행렬을 적용해 복사한다. 이는 참조연산이 생성한 공구경로(tool path)를 검색된 형상의 위치로 복사하는 효과가 있다.

Fig. 4는 제안한 방법에 대한 개념을 보여준다. 작업자는 Fig. 4-①에 해당하는 공구경로를 생성하기 위해 한 개의 가공연산을 정의한다. 이 때 Fig. 4-②의 원형 모서리를 가공연산을 정의하는 데 참조한다. 그리고 Fig. 4-①을 참조연산으로 지정하고, Fig. 4-②를 참조형상으로 지정해, Fig. 4-③과 같이 참조형상과 동일한 형상들을 검색한다. 그리고, Fig. 4-②에서 Fig. 4-③의 각 원형 형상으로 이동시키는 변환행렬을 계산한다. 최종적으로, 참조연산에 변환행렬을 정의해 Fig. 4-①의 공구경로를 Fig. 4-③의 각 형상에 대응하는 위치로 복사한다.

제안한 방법에서는 참조연산과 참조형상을 선택하는 과정만 사용자가 개입하고, 나머지 과정은 자동으로 수행된다. 따라서 동일한 가공형상에 대해서 한 번의 가공연산만 정의하면 나머지 형상에 대해서는 미리 정의된 가공연산을 자동으로 복사해 사용할 수 있다. 기존의 CAM 소프트웨어들은

정의된 가공연산을 복사해 사용하는 기능을 제공한다. 그러나 이 기능을 이용하기 위해서는 사용자가 변환행렬을 직접 정의해 주거나, 복사할 위치를 일일이 지정해 주어야 한다. 더군다나 복사 위치가 규칙적이지 않다면 위치를 정의하는데 매우 긴 시간이 걸린다.

### 3.2 형상 검색 방법

Fig. 3의 두 번째 단계에서는 참조형상과 동일한 형상을 검색한다. 동일한 형상을 검색하기 위한 다양한 방법들이 제안되었고, 이에 대한 내용이 참고문헌 (18)에 잘 정리되어 있다. 형상을 검색하기 위한 대표적인 방법에는 가공 특징형상 인식 기반 방법(manufacturing feature recognition-based method), 제품 정보 기반 방법(product information-based method), 3차원 객체 인식 기반 방법(3D object recognition-based method), 그래프 기반 방법(graph-based method), 히스토그램 기반 방법(histogram-based method), 전역 특징 기반 방법 등이 있다.

본 연구에서 제안하는 방법에서는 전체 형상에서 참조형상과 동일한 면 또는 모서리의 집합으로 구성된 부분 형상을 찾는 것이 필요하다. 이 관점에서 판단해 보면, 가공 특징형상 인식 기반 방법과 제품 정보 기반 방법, 3차원 객체 인식 기반 방법 및 그래프 기반 방법은 적용할 수 없다. 가공 특징형상 인식 기반 방법은 전체 형상을 특징형상 인식 기술로 인식된 가공 특징형상들로 표현한 후, 두 형상을 구성하는 가공 특징형상들을 서로 비교해 형상을 검색하는 방법이다. 그러나 특징형상은 참조형상보다 더 큰 단위의 형상 집합이기 때문에 본 연구에는 적용할 수 없다. 제품 정보 기반 방법은 제품을 표현하는 다양한 정보들을 비교해 형상을 검색하는 방법이다. 이 방법도 CAD 모델 단위의 검색만 가능하기 때문에 본 연구에는 적용할 수 없다. 3차원 객체 인식 기반 방법은 3차원 형상의 2차원 이미지들을 비교해 형상을 검색하는 방법으로, 역시 본 연구에는 적용할 수 없다. 그래프 기반 방법은 전체 형상의 특징을 그래프로 표현한 다음, 그래프 비교를 통해 형상을 검색하는 방법이다. 형상의 특징은 경계표현 그래프(boundary representation graph), 리브 그래프(Reeb graph), 골격 그래프(skeletal graph) 등으로 표현할 수 있다. 이 방법 또한 전체 형상 비교를 통해 검색하는 방법으로 부분 형상을 검색하는 데에는 적합하지 않다. 히스토그램 기반 방법은 형상의 곡면 위에 있는

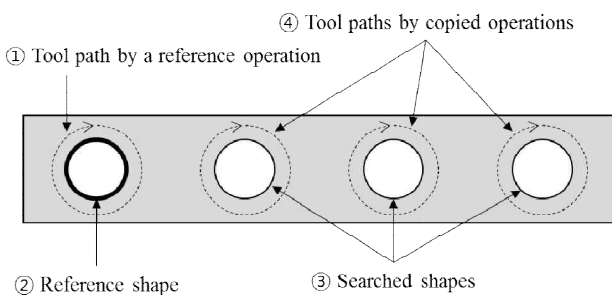


Fig. 4 Concept of the proposed method



비교를 통해 참조형상과 동일한 형상을 간단히 찾아낼 수 있다.

참조형상이 여러 개가 주어졌을 경우에는 참조형상 각각에 대해 동일한 형상을 검색한다. 이 경우, 참조형상 각각의 면 또는 모서리에 대응하는 검색한 형상들과의 연관관계가 저장되어야 한다.

### 3.3 변환행렬 계산

참조형상과 동일한 면 또는 모서리를 검색하고 나면, 참조형상을 검색한 면 또는 모서리의 위치로 복사하기 위한 변환행렬을 계산한다. 이는 Fig. 2의 세 번째 단계에 해당한다. 본 연구에서는 제트엔진 형상만을 고려하기 때문에, 모든 변환은 축 회전변환으로 가정한다. 작업자가 제트엔진의 실제 CAM 모델을 수작업으로 정의할 때에도, 유사한 부분의 가공연산을 축 회전복사를 이용해 정의하므로, 이러한 가정은 무리가 없다. 회전축은 사용자가 CAD 모델에서 모서리나 원통면 선택 등을 통해 직접 정의한다.

우선 각각의 참조형상으로부터 검색된 면 또는 모서리까지의 회전변환을 계산한다. 이 때, 회전변환은 참조형상의 무게중심(Fig. 5의  $\mathbf{p}_r$ )과 검색된 형상의 무게중심(Fig. 5의  $\mathbf{p}_s$ )을 기준으로 계산한다. 회전변환을 정의하는 것이 가능 하려면 두 가지 조건을 만족시켜야 한다:

- 1) 회전축에서부터 참조형상까지의 거리  $d_r$  과  $d_s$ 는 동일해야 한다.
- 2) 두 무게중심을 연결한 벡터(Fig. 4의  $\mathbf{v}_{sr}$ )가 회전축과 수직이어야 한다.

만약 이 두 조건을 만족하지 않는다면 검색된 형상은 제외된다. 이 때, 3.2절에서 검색된 형상 중 참조형상과 무게중심이 다르거나, 회전변환 적용이 불가능한 형상은 추가적으로 걸러진다. 두 조건을 만족하면 세 점  $\mathbf{p}_a$ ,  $\mathbf{p}_r$ ,  $\mathbf{p}_s$ 로부터 회전각  $\theta$

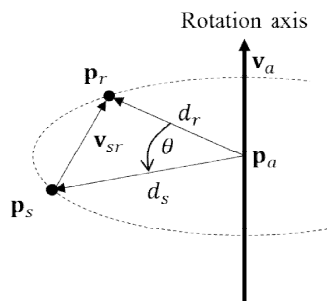


Fig. 5 Defining an axis rotation from the center points and the axis

를 벡터의 내적을 이용해 다음과 같이 계산할 수 있다.

$$\theta = \cos^{-1} \left[ \frac{(\mathbf{p}_r - \mathbf{p}_a) \cdot (\mathbf{p}_s - \mathbf{p}_a)}{\|\mathbf{p}_r - \mathbf{p}_a\| \|\mathbf{p}_s - \mathbf{p}_a\|} \right]. \quad (1)$$

이 때,  $\mathbf{p}_a$ 는 두 점  $\mathbf{p}_r$ 과  $\mathbf{p}_s$ 를 통과하는 직선으로부터 가장 가까운 회전축 위의 점이다.

회전축과 회전각을 계산하면 이로부터 변환행렬을 계산할 수 있다. 회전축의 방향을 단위벡터  $\mathbf{v}_a = (u, v, w)$ , 회전축 위의 한 점을  $\mathbf{p}_a = (a, b, c)$ , 회전각을  $\theta$ 라고 하면, 축 회전변환  $\mathbf{T}$ 는 다음과 같이 3행 4열의 행렬로 표현된다.

$$\mathbf{T} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}. \quad (2)$$

여기서,

$$r_{11} = u^2 + (v^2 + w^2) \cos \theta, \quad (3)$$

$$r_{12} = uv(1 - \cos \theta) - w \sin \theta, \quad (4)$$

$$r_{13} = uw(1 - \cos \theta) + v \sin \theta, \quad (5)$$

$$r_{21} = uv(1 - \cos \theta) + w \sin \theta, \quad (6)$$

$$r_{22} = v^2 + (u^2 + w^2) \cos \theta, \quad (7)$$

$$r_{23} = vw(1 - \cos \theta) - u \sin \theta, \quad (8)$$

$$r_{31} = uv(1 - \cos \theta) - v \sin \theta, \quad (9)$$

$$r_{32} = vw(1 - \cos \theta) + u \sin \theta, \quad (10)$$

$$r_{33} = w^2 + (u^2 + v^2) \cos \theta, \quad (11)$$

$$t_x = (a(v^2 + w^2) - u(bv + cw)) \times (1 - \cos \theta) + (bw - cv) \sin \theta, \quad (12)$$

$$t_y = (b(u^2 + w^2) - v(cu + aw)) \times (1 - \cos \theta) + (cu - av) \sin \theta, \quad (13)$$

$$t_z = (c(u^2 + v^2) - w(cu + bv)) \times (1 - \cos \theta) + (av - bu) \sin \theta. \quad (14)$$

각각의 참조형상에 대해서, 검색된 면이나 모서리까지의 회전변환을 계산하고 나면, 동일한 회전변환이 참조형상의 개수만큼 존재해야 한다. 만약 참조형상의 개수와 다르다면 이는 동일한 형상이 아니라고 판단한다. 회전변환의 개수를 확인하고 나면, 중복된 회전변환을 제거한다. 그리고 회전변환을 회전각을 기준으로 정렬하여, 가공연산이 회전축을 따라 순서대로 복사되도록 한다.

회전변환을 계산하고 하면, Fig. 3의 마지막 단계로, 회전변환을 참조연산에 적용해 회전복사를 한다. 이 부분은 CAM 소프트웨어의 API를 이용해 구현할 수 있다.

4. 구현 및 실험

본 연구에서 제안한 방법을 프로토타입 프로그램 구현을 통해 확인하였다. CAM 소프트웨어로는 CATIA의 CAM 모듈을 이용했다. Fig. 6은 프로토타입 프로그램의 구조를 보여준다. OpCopier 모듈은 사용자를 위한 GUI를 제공하며 전체 시스템을 제어한다. ShapeFinder 모듈은 본 연구의 핵심 모듈로 형상 검색 및 변환행렬 계산을 처리한다. ShapeFinder 모듈은 상용 CAD 소프트웨어인 CATIA로부터 IGES 파일 형식으로 전체 형상에 대한 CAD 모델과 참조형상을 받고, OpCopier 모듈로부터 회전축 정보를 받는다. 그리고 계산한 변환행렬을 OpCopier에 돌려주고, OpCopier는 CATIA Automation API를 이용해 참조연산을 복사한다. ShapeFinder 모듈은 공개용 형상모델링 커널

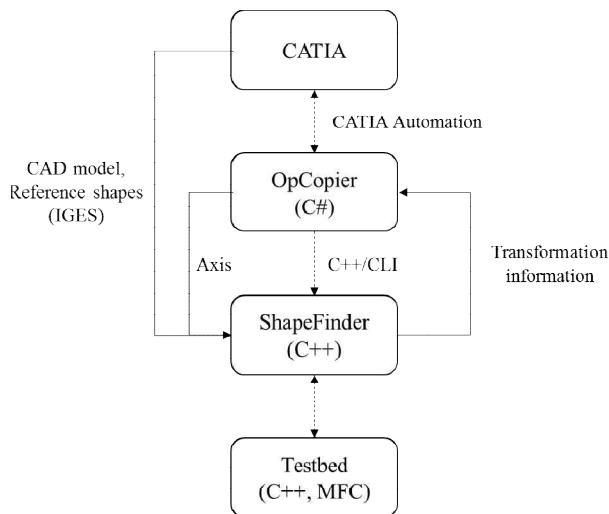


Fig. 6 Architecture of the prototype program

인 OpenCascade<sup>(25)</sup>를 이용해 구현하였다. 그리고 ShapeFinder 모듈의 테스트 및 디버깅을 위해 Testbed 프로그램을 구현하였다. Fig. 7은 OpCopier의 실행 화면을 보여준다.

실험은 인텔 i7 CPU 및 8GB 램을 가진 PC에서 진행이 되었다. Fig. 8은 실험을 위해 사용한 테스트 모델이다. 이 모델은 303 개의 면과 816 개의 모서리를 가지고 있다. 이 모델은 테스트를 위해 인위적으로 만들어진 형상이다.

우선 윗부분에 있는 슬롯(slot) 형상에 대해 테스트 해 보았다. 먼저 Fig. 9(a)의 점선 원 부분의 슬롯 형상을 가공하기 위한 가공연산을 CATIA의 Profile Contouring 기능을 이용해 정의했다. 그리고 OpCopier를 이용해 참조형상과 회전축을 지정했다. 참조형상은 Fig. 9(b)에 굵은 실선으로 표시한 두 모서리이다. 참조연산으로 가공할 슬롯 형상은 Fig. 9(a)에서 보면 원통면이 두 개의 면으로 표현된다. 그러나 다른 슬롯 형상에는 원통면이 한 개의 면으로 표현된다. 이러한 경우 두 개의 면을 참조형상으로 지정하면 슬롯 형상들을 찾을 수 없기 때문에 모든 슬롯 형상들이 공통으로 가지고 있는 두 개의 모서리를 참조형상으로 지정했다.

OpCopier를 이용해 슬롯 형상을 검색하고 Profile Contouring을 복사하는 데 약 0.5초가 걸렸다. Fig. 9(b)의 굵은 실선들은 OpCopier를 이용해 검색한 형상을 나타낸다. 또한 가는 실선들은 Profile Contouring을 복사해 생성한 공구경로들이다. 이 예제의 경우에는 참조형상으로 두 개의 모서리를 지정했으며, 가공형상의 위상정보가 다른 경우에도 적용할 수 있다는 것을 보여준다.

동일한 모델에 대해서, 모따기(chamfer) 형상에

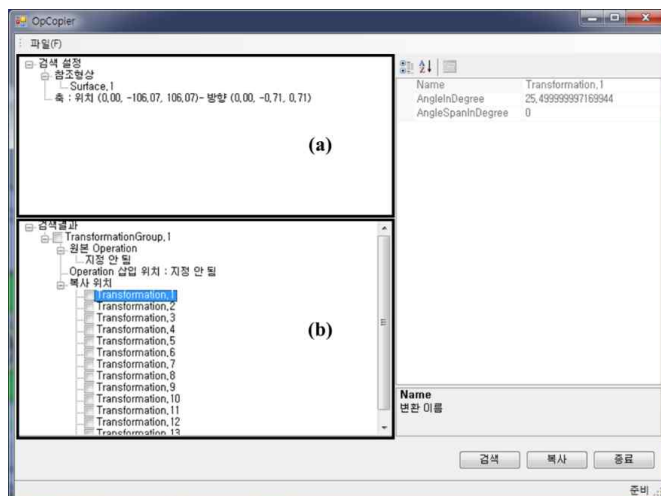


Fig. 7 Screenshot of OpCopier: (a) input data - reference shapes and rotation axis, and (b) computed transformations

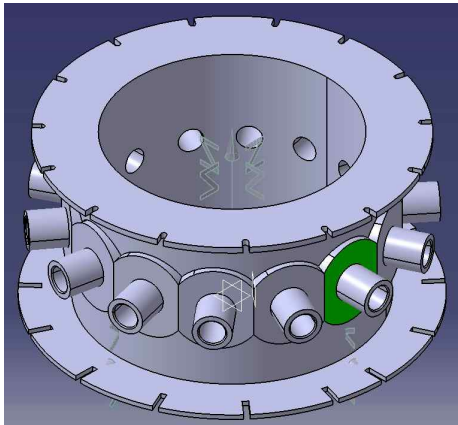
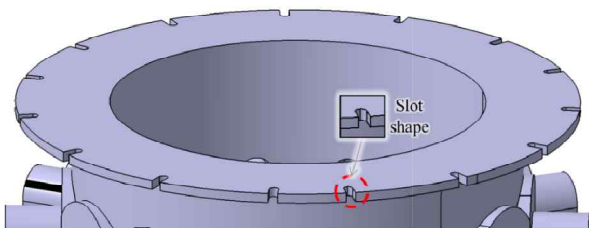
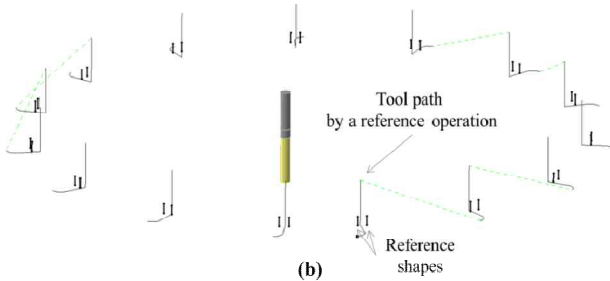


Fig. 8 Test model



(a)

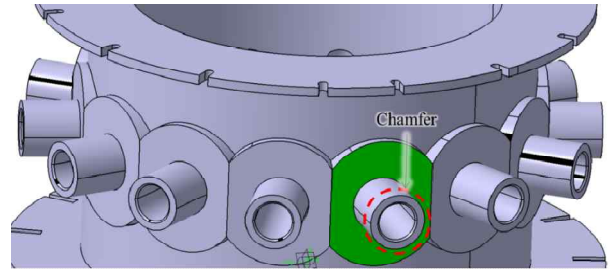


(b)

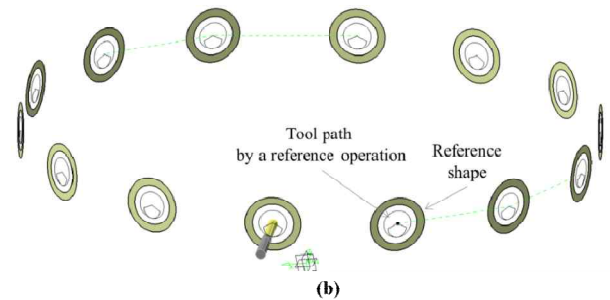
Fig. 9 Slot test: (a) slot shape, and (b) copied operations (CATIA model is hidden)

대해 테스트를 해 보았다. 먼저 Fig. 10(a)의 점선 원 부분의 모따기 형상을 가공하기 위한 가공연산을 CATIA의 Sequential Groove 기능을 이용해 정의했다. 그리고 OpCopier를 이용해 Fig. 10(b)의 면을 참조형상으로 지정했다. OpCopier를 이용해 모따기 형상을 검색하고 Sequential Groove를 복사하는 데 약 0.5초가 걸렸다. Fig. 10(b)에 검색한 형상이 나타나 있다. 또한 가는 실선들은 참조연산을 복사해 생성한 공구경로들이다.

복잡한 형상에 대해서도 적용 가능함을 보여주기 위해 실제 제트엔진 압축기 케이스 모델에 적용해 보았다. 업체의 보안 문제로 테스트 모델의 전체 형상을 보여줄 수는 없지만, 제트엔진 압축기 케이스는 Fig. 2의 형상보다 더 복잡하다. 본 연구에서 사용한 모델은 GE가 설계한 GENx Forward Case<sup>(26)</sup>의 CAD 모델이다. 이 모델은



(a)



(b)

Fig. 10 Chamfer test: (a) chamfer shape, and (b) copied operations (CATIA model is hidden)

18,139개의 면과 60,712개의 모서리를 가지고 있다. 이 모델은 실제 설계 모델이며 실험을 위한 어떠한 인위적인 가공도 하지 않았다.

첫 번째로 구멍 형상에 대해 테스트 해 보았다. 먼저 Fig. 11(a)의 점선 원 부분의 구멍 형상을 가공하기 위한 가공연산을 CATIA의 Drilling 기능을 이용해 정의했다. 그리고 OpCopier를 이용해 원호 모양의 모서리를 참조형상으로 지정했다. OpCopier를 이용해 구멍 형상을 검색하고 Drilling 연산을 복사하는 데 약 6초가 걸렸다. Fig. 11(b)의 굵은 실선들은 OpCopier를 이용해 검색한 형상을 나타낸다. 또한 가는 실선들은 참조연산을 복사해 생성한 공구경로들이다.

두 번째로 보스(boss)의 외곽 가공에 대해 테스트 해 보았다. 보스 외곽 가공은 CATIA의 Profile Contouring 기능으로 정의했다. Fig. 12에서처럼, 보스 형상의 윗면을 참조형상으로 선택하여 검색하였다. 형상을 검색하고 참조연산을 복사하는데 약 6초가 걸렸다. 검색된 형상과 참조연산을 복사해 생성한 공구경로가 Fig. 12에 표시되어 있다. 이 두 실험은 제안한 방법이 실제로 사용하는 형상에 대해서도 적용할 수 있다는 것을 보여준다.

GENx Forward Case에 대해 본 연구에서 제안한 방법을 이용해 전체 공정에 대해 적용해 보았다. CAD 모델로부터 CAM 모델을 정의하고, 가공경로를 생성한 후, 실제 가공이 끝날 때까지 8개월이 소요된다. 이 중, CAM 모델을 정의하고 가공경로를 생성하는 데 걸리는 기간이 20일이다. 본 연



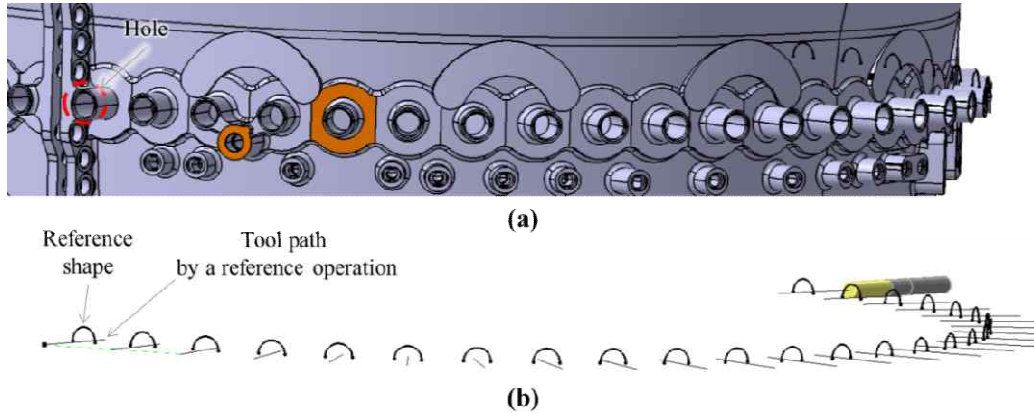


Fig. 11 Hole test: (a) hole shape, and (b) copied operations (CATIA model is hidden)

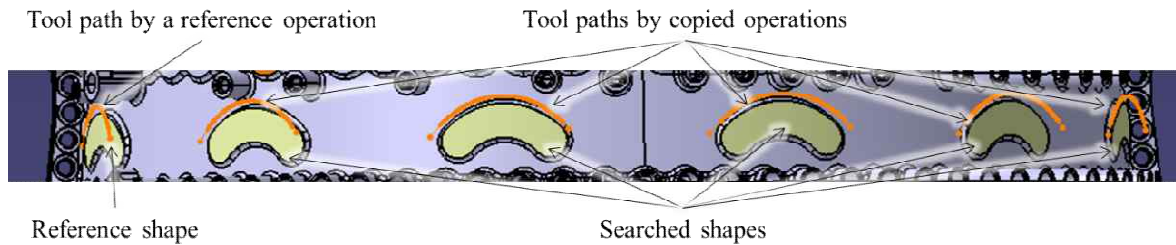


Fig. 12 Boss machining test

구에서 제안한 방법을 적용할 경우, 이 20일의 기간이 12일로 줄어들어 8일의 기간을 단축시킬 수 있었다. 이는 가공경로 생성 기간의 40%에 해당한다.

### 5. 결론

본 연구에서는 CAD 모델로부터 CAM 모델을 생성하기 위한 새로운 방법을 제안하였다. 이 방법에서는 사용자가 정의한 참조연산을 동일한 가공형상을 가지는 곳에 복사해 사용한다. 동일한 가공형상을 찾기 위해 전역 특징 기반 형상 검색 방법을 사용한다. 그리고 제안한 방법의 검증을 위해 두 가지 모델에 대해 실험을 하였고, 만족스러운 결과를 보여주었다. 본 연구에서 제안한 방법은 실제 현장에 적용 중이며, 작업 시간을 크게 단축시켜 주고 있다.

### 후 기

이 논문은 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(NRF-2014R1A1A1006181)입니다.

### 참고문헌 (References)

- (1) StandardAero Components homepage, <http://standardaero-components.com/>.
- (2) Han, J. H., Pratt, M. and Regli, W. C., 2000, "Manufacturing Feature Recognition from Solid Models," *IEEE Trans. Rob. Autom.*, Vol. 16, No. 6, pp. 782~796.
- (3) Shah, J. J., Anderson, D., Kim, Y. S. and Joshi, S., 2001, "A Discourse on Geometric Feature Recognition from CAD Models," *J. Comput. Inf. Sci. Eng.*, Vol. 1, No. 1, pp.41~51.
- (4) Kim, B. C. and Song, I., 2015, "Automatic Generation of CAM Model for Machining Holes for Jet Engine Compressor Case Based on Feature Recognition," *Trans. Korean Soc. Mech. Eng. A*, Vol. 39, No. 3, pp.337~345.
- (5) Joshi, S. and Chang, T. C., 1988, "Graph Based Heuristics for Recognition of Machined Features from a 3-D Solid Model," *Comput. Aided Des.*, Vol. 20, No. 2, pp. 58~66.
- (6) Chuang, S. H. and Henderson, M. R., 1990, "Three-Dimensional Shape Pattern Recognition Using Vertex Classification and the Vertex-Edge Graph," *Comput. Aided Des.*, Vol. 22, No. 6, pp. 377~387.

- (7) Gavankar, P. and Henderson, M. R., 1990, "Graph-Based Extraction of Protrusions and Depressions from Boundary Representations," *Comput. Aided Des.*, Vol. 22, No. 7, pp. 442~450.
- (8) Gao, S. and Shah, J. 1998, "Automatic Recognition of Interacting Machining Features Based on Minimal Condition Subgraph," *Comput. Aided Des.*, Vol. 30, No. 9, pp. 727~739.
- (9) Tang, K. and Woo, T., 1991, "Algorithmic Aspects of Alternating Sum of Volumes. Part I: Data Structure and Difference Operation," *Comput. Aided Des.*, Vol. 23, No. 5, pp. 357~366.
- (10) Kim, Y. S. and Wilde, D. J., 1992, "A Convergent Convex Decomposition of Polyhedral Objects," *J Mech. Des. N. Y.*, Vol. 114, No. 3, pp. 468~477.
- (11) Sakurai, H., 1995, "Volume Decomposition and Feature Recognition, Part I: Polyhedral Objects," *Comput. Aided Des.*, Vol. 27, No. 11, pp. 833~843.
- (12) Sakurai, H. and Dave, P., 1996, "Volume Decomposition and Feature Recognition, Part II: Curved Objects," *Comput. Aided Des.*, Vol. 28, No. 6~7, pp. 519~537.
- (13) Woo, Y., 2003, "Fast Cell-Based Decomposition and Applications to Solid Modeling," *Comput. Aided Des.*, Vol. 35, No. 11, pp. 969~977.
- (14) Kim, B. C. and Mun, D., 2014, "Feature-Based Simplification of Boundary Representation Models Using Sequential Iterative Volume Decomposition," *Comput. Graph.*, Vol. 38, pp. 97~107.
- (15) Kim, B. C. and Mun, D., 2014, "Stepwise Volume Decomposition for the Modification of B-Rep Models," *Int. J. Adv. Manuf. Tech.*, Vol. 75, No. 9-12, pp. 1393~1403.
- (16) Vadenbrande, J. H. and Requicha, A. A. G., 1993, "Spatial Reasoning for the Automatic Recognition of Machinable Features in Solid Models," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 15, No. 12, pp. 1269~1285.
- (17) Regli, W. C., Gupta, S. K. and Nau, D. S., 1995, "Extracting Alternative Machining Features: An Algorithmic Approach," *Res. Eng. Des.*, Vol. 7, No. 3, pp. 173~192.
- (18) Iyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y. and Ramani, K., 2005, "Three-Dimensional Shape Searching: State-of-the-Art Review and Future Trends," *Comput. Aided Des.*, Vol. 37, No. 5, pp. 509~530.
- (19) Elad, M., Tal, A. and Ar, S., 2001, "Content Based Retrieval of VRML Objects-An Iterative and Interactive Approach," *In Proc. of Eurographics Multimedia Workshop 2001*, pp. 97~108.
- (20) Rea, H. J., Corney, J. R., Clark, D. E. R., Pritchard, J., Breaks, M. L. and MacLeod, R. A., 2002, "Part-Sourcing in a Global Market," *Concurrent Eng-Res. A.*, Vol. 10, No. 4, pp. 325~333.
- (21) Vranic, D., Saupe, D. and Richter, J., 2001, "Tools for 3D Object Retrieval: Karhunen-Loeve Transform and Spherical Harmonics," *In Proc. of the IEEE Workshop on Multimedia Signal Processing 2001*, pp. 293~298.
- (22) Saupe, D. and Vranic, D., 2001, "3D Model Retrieval with Spherical Harmonics and Moments," *In Proc. of the DAGM 2001*, pp. 392~397.
- (23) Tanaka, K., Sano, M., Mukawa, N. and Kaneko, H., 1993, "3D Object Representation Using Spherical Harmonic Functions," *In Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1873~1880.
- (24) Kazhdan, M., Funkhouser, T. and Rusinkiewicz, S., 2003, "Rotation Invariant Spherical Harmonic Representation of 3D Shape Descriptors," *In Proc. of the ACM/Eurographics Symposium on Geometry Processing 2003*, pp. 167~175.
- (25) OpenCascade homepage, <http://www.opencascade.org/>.
- (26) GE Aviation homepage, <http://www.geaviation.com/commercial/engines/genx/>.