

특징형상인식을 이용한 가공데이터 추출에 관한 연구 A Study on Machining data Extraction using Feature Recognition Rules

이 석 회(부산대학교)
정 구 섭*(진주전문대학)

Abstract

This paper presents a feature recognition system for recognizing and extracting feature information needed for machining from design data contained in the CAD database of AutoCAD system. The developed system carries out feature recognition from an orthographic view of a press mold containing not only atomic features such as holes, pockets, and slots, but also compound features. Based on the result of feature recognition, it generates a 3-D modeling of the press mold. Especially, The feature recognition part is designed for detecting feature styles according to feature definition and classification, extracting parameters for various atomic features, and constructing necessary data structures for the recognized features.

1. 서론

최근의 금형제품의 설계 및 생산에 있어서는 다품종 소량 생산체제로 접어 들었으며 제품의 형상이 복잡, 다양해지고 생산 수명 주기(Production Life Cycle)가 단축되는 등 복잡한 문제에 대처하기 위하여 자동화된 CAD/CAM 시스템이 절실히 요구되고 있다. CAD에서 생성된 도면정보를 가공정보로 자동으로 변환하기 위해서는 먼저 CAD시스템에서 작성된 도면을 자동으로 인식할 수 있어야 하며 이 인식된 결과에 의해 생성된 형상인식 데이터를 이용하여 가공정보를 만든다. 따라서 형상인식 과정이 무엇보다 중요하다. 하지만 CAD 데이터는 기본적인 요소인 점(point), 직선(line), 원호(arc), 원(circle)들로 이루어져 있기 때문에 생산자동화 측면에서는 그 데이터를 직접 이용하기에는 적합하지 않다. 예를들면 기계가공을 위한 공정설계의 경우에 필요한 제품 정보는 구멍(hole)이라든지 포켓(pocket), 슬롯(slot)등과 같은 특징형상(feature)이지 직선이나 원호와 같은 기본 entity 정보가 아니다. 따라서 CAD/CAM 인터페이스를 위해서는 형상인식 과정은 필수적이며, 이러한 기본 entity정보를 이용하여 구멍, 포켓, 슬롯과 같은 가공 특징형상들을 인식하기 위해서는 형상인식에 관련된 지식베이스(Knowledge base)와 설계도면의 형상 데이터베이스를 이용하여 특징형상 인식규칙을 만든다. 이러한 특징형상 인식을 하기 위한 다양한 연구들이 진행 되어 왔다.^{[1][2][3]}

본 연구에서는 CAD 시스템의 데이터 구조로부터 가공 특징형상을 인식하고 추출할때 DXF(Drawing Interchange Files)나 IGES(Initial Graphics Exchange Specification)데이터 파일 등의 변환파일을 이용하지 않고 CAD 데이터 베이스내의 엔티티 속성 데이터를 이용하여 화면상에서 직접 형상인식을 수행하였다. 형상인식의 대상으로서는 비회전형상

들을 대상으로 프레스금형 부품에서 가장 많이 사용되고 있는 기본특징형상들과 이들의 조합인 복합형상에 대하여 개별적인 알고리즘을 정립하여, 2차원으로 그려진 삼면도를 입력 드로잉(drawing) 파일로 삼아 형상인식을 수행하였다. 여기서 인식된 각 특징형상들의 추출데이터를 이용하여 3차원 형상모델링과, NC가공 데이터로도 이용하게 하여 CAD/CAM의 통합화로 활용될 수 있도록 하였다.

본 시스템의 특징형상 인식 부분은 AutoCAD환경의 AutoLISP을 사용하여 프로그래밍하였고, 여기서 나온 인식 결과 데이터를 이용한 3차원 형상모델링 작업은 ADS(AutoCAD Development System)환경의 C언어와 ACAD에서 제공하는 3차원 솔리드 모델러인 AME(Advanced Modeling Extension)를 사용하여 수행하였다.

2. 특징형상 인식 시스템 2.1 시스템의 전체 흐름도

본 시스템의 전체적인 흐름은 Fig. 1에서와 같이 ACAD시스템에서 생성된 2차원으로 그려진 삼면도를 특징형상 인식을 수행하기 위한 도면으로 입력하면, ACAD 드로잉 데이터베이스(drawing database)에서는 입력도면에 대한 각 기본 엔티티(entity)들의 데이터베이스를 생성한다. 이러한 기본 엔티티 정보와 형상들의 기하학적, 위상학적 관계 등 형상인식에 관련되는 지식베이스를 이용하여 특징형상 인식 알고리즘을 만들고, 정립된 알고리즘대로 특징형상 인식을 수행하면 특징형상 데이터 구조에 맞게 인식 결과 데이터를 추출한다. 추출된 결과 데이터를 사용하여 일련의 과정을 거쳐 3차원 솔리드 모델링을 수행함으로써 기본 특징형상들의 파라미터들과 계층관계를 확인할 수 있도록 하였다.

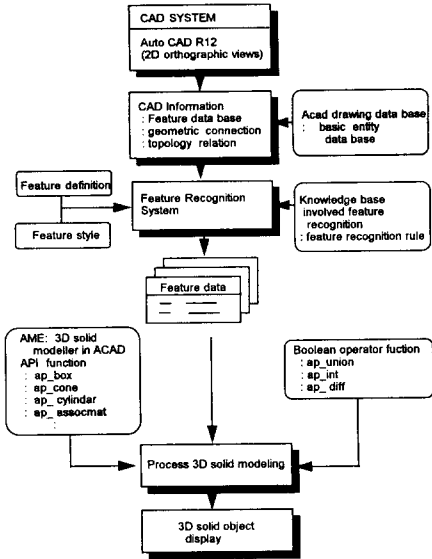


Fig. 1 The system flowchart

2.2 특징형상 데이터 구조

기본 특징형상을 정의하는데 필요한 데이터는 모든 기본 특징형상들이 공통적으로 필요로 하는 데이터들과 그렇지 않는 데이터로 기본 특징형상들을 표현하였다. 기본 특징형상들이 공통적으로 갖고 있는 데이터는 Fig. 2와 같이 특징형상의 이름(feature name), 특징형상의 중심좌표(center point) 가공깊이(machining depth), 가공방향(machining direction), 직육면체형상(box)에 필요한 길이(length), 폭(width), 높이(height)가 있다. Fig. 2는 막힌 사각포켓과 막힌구멍에 대해서 공통적인 데이터를 가지는 것을 보여주고 있고, Fig. 3은 본 연구에서 형상인식이 수행된 특징형상 중에서 구멍류의 기본특징형상 데이터구조를 보여주고 있다.

3. 특징형상 및 형상인식 알고리즘

3.1 특징형상의 분류

특징형상의 개념은 도면이 표현하고 있는 부품의 모양, 위치, 크기, 방향의 기하학적 정보등의 미리 정의된 특징형상들의 조합으로 표현함으로써, CAD/CAM 시스템이 동일한 형태로 그 정보를 공유할 수 있도록 하는데 이 특징형상의 목적이 있다.^[4] 따라서 특징형상은 설계측면으로만 모델링된 CAD 데이터로부터 가공작업에 필요한 정보를 도출하기 위해서는 가공작업에 의하여 생성되는 형상들을 일정한 형태로 정의할 필요가 있다. 본 연구에서는 비회전형상들을 대상으로 금형부품의 드릴링, 밀링 및 보링 등의 절삭가공 작업에 의하여 생성될 수 있는 형상들을 고려하여 Fig. 4와 같이 특징형상들을 정의, 분류하였다.

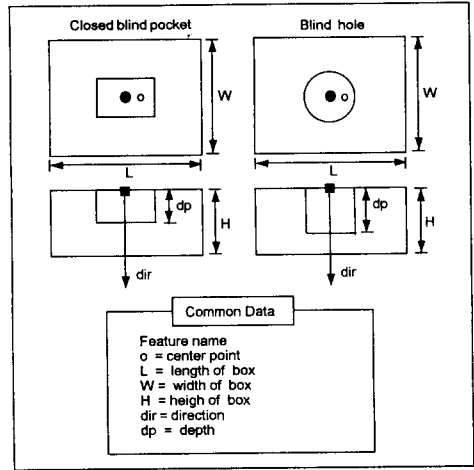


Fig. 2 Common data of atomic features for definition

Feature style	Feature data structure
	Blind hole BHO (feature name) o (x_c, y_c, z_c) = center point x_c = center x coordinate y_c = center y coordinate z_c = center z coordinate r = radius dp = depth dir = direction
	Counterbore CB (feature name) o (x_c, y_c, z_c) = center point r1 = radius 1 r2 = radius 2 dp1 = depth 1 dp2 = depth 2 dir = direction
	Countersink CS (feature name) o (x_c, y_c, z_c) = center point r1 = radius 1 r2 = radius 2 dp1 = depth 1 dp2 = depth 2 dir = direction
	Through hole THO (feature name) o (x_c, y_c, z_c) = center point r = radius dp = depth dir = direction

Fig 3(a) Data structure of hole features

3.2 형상인식 알고리즘

형상인식을 수행하기 위해서는 먼저 기본적인 형상들에 대하여 형상특징을 정의해 놓고, 정의해 놓은 기본형상들에 대하여 각각의 개별적인 인식 알고리즘을 만든 후, 3각법에 의하여 그려진 각 투상도를 인식 규칙에 따라 좌표변환을 수행하여 새로운 좌표계를 부여한다. 그리고 난 다음 삼면도 도면 인식과 특징형상 인식을 수행한다. 형상인식은 크게 두 부분으로 나누어 수행되는데 삼면도 도면의 인식과 삼면도 내에 포함되어 있는 특징형상 인식으로 나눌 수 있다. 삼면도 도면의 인식은 정면도, 평면도, 측면도를 검색하여 각 단면도의 기준점 b1, b2, b3을 찾고, 각 단면도들간에 기하학적 상관관계를 확인하여 직육면체 형상(box)에 필요한 길이, 폭, 높이를 알아낸다. 그런 다음 각 단면도내에 있는 특징형상들

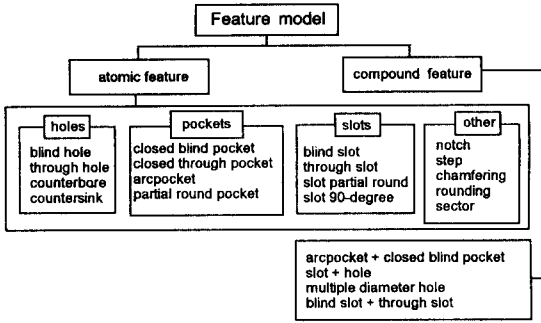


Fig. 4 Classification of features

을 이미 정의해 놓은 인식규칙에 따라 형상인식을 수행한다.

3.3 삼면도 인식 알고리즘

삼면도 인식과정을 설명하기 위해서 Fig. 5의 관통구멍이 포함되어 있는 투상도를 예를 들어 살펴보면 다음과 같다.

[step 1] 전체도면에서 좌표변환에 의해 삼면도로 분리한다. 이때 인식의 편의를 위해서 삼면도의 분리점을 입력하는 방식을 취하여 삼면도 중앙에 기준점(reference point) P를 잡는다. 기준점 P보다 절대좌표 X, Y가 작은 도면요소를 정면도(front view)가 존재하는 영역으로 설정하고, 기준점 P보다 X좌표값은 작고 Y좌표값은 큰 위치에 있는 도면요소를 평면도(top view)를 두고, 기준점 P보다 X좌표값은 크고 Y좌표값은 작은 도면요소를 우측면도(side view)로 설정하여 새로운 좌표계를 부여한다.

[step 2] 정면도에서 기준점 P보다 절대좌표 X, Y점이 작은 모든 엔티티를 검색하여 그 중에서 가장 최소인 좌표점을 선택하여 정면도의 기준점인 b1를 찾는다.

[step 3] 측면도에서 기준점 P의 X좌표값보다 큰 엔티티를 모두 검색하여 엔티티들의 시작점과 끝점 중 가장 최소가 되는 점을 찾아 측면도의 base point인 b2를 선정한다.

[step 4] 평면도에서는 기준점 P의 절대좌표 Y값보다 큰 엔티티를 모두 검색하여 가장 최소가 되는 좌표점을 찾아 평면도의 기준점인 b3를 선정한다.

[step 5] 각 단면도의 기준점 b1, b2, b3의 좌표값들을 화면에 출력하고 각 단면도들간에 기하학적 상관관계를 확인하여 직육면체 형상에 필요한 길이, 폭, 높이를 알아낸다.

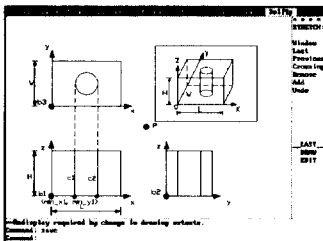


Fig. 5 3-view drawing and 3d solid object of through hole

3.4 특징형상 인식 알고리즘

3.4.1 관통구멍과 막힌구멍 인식 규칙

구멍형상의 특징은 평면도내에 원의 형상이 있고, 이와 관련하여 다른 투상도에 원의 좌표와 관련이 있는 선이 나타난다. 이 원의 중심좌표와 지름은 드로잉 데이터 베이스에서 추출한다. 투상도에 원이 포함되어 있는 형상은 관통구멍, 막힌구멍, 카운터보어, 카운터싱크 중의 하나가 될 수 있는데, 동일한 중심을 갖는 다른 원이 있는지를 검색하여 동일한 중심을 갖는 원이 존재하지 않으면 관통구멍이나 막힌구멍으로 인식하고, 동일한 중심을 갖는 다른 원이 존재하면 카운터보어나 카운터싱크로 인식한다. 다음은 원이 검색된 이외의 투상도에서 원과 관련된 해당 line이 존재하는지를 확인하여 조건을 만족하면 Hole로 인식한다. 관통구멍과 막힌구멍의 인식 구분은 Fig. 5에서 보는 바와 같이 드로잉 데이터 베이스에서의 기준점 b1의 z 좌표값 min_z1 과 원과 관련된 해당 line entity C1, C2의 한쪽 끝점의 z 좌표값이 일치하면 관통구멍으로 인식하고, 반면 C1, C2의 양 끝점의 z 좌표값이 기준점 b1의 z 좌표값보다 크면 막힌구멍으로 인식한다.

3.4.2 카운터보어와 카운터싱크 인식규칙

Fig. 6에 나타난 것과 같이 카운터보어 및 카운터싱크의 형상의 특징은 하나의 투상도에 중심점이 같은 두 개의 동심원이 존재하는 태에 있다. 카운터보어와 카운터싱크를 인식하기 위해서는 먼저 한 투상도에 중심이 같은 두개의 동심원이 존재하는지의 여부를 검색한 다음, 두 원의 가장자리의 좌표값을 찾은 후에 이 좌표값에 연관된 직선 1, 2, 1', 2'가 다른 투상도에 존재하는지를 확인한다. 이 조건을 만족하면 직선 3, 4, 5 또는 3', 4', 5'를 조사한다. 카운터보어 형상은 직선 1, 2의 양 끝점은 큰 원의 가장자리 좌표와 일치하는 반면에 카운터싱크는 직선 1', 2'의 한쪽 끝은 큰 원의 가장자리에 존재하고 다른 끝은 작은 원의 가장자리에 존재한다.

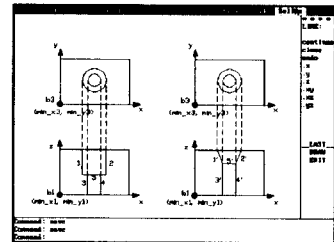


Fig. 6 3-view drawing of counterbore and countersink

3.4.3 관통사자포켓 및 막힌사자포켓 인식규칙

포켓류의 형상은 평면도내의 형상이 대부분 폐루프(Closed Loop)를 이루고 있고 루프의 형상에 따라 대부분의 특징형상이 정의 된다. 구멍류의 형상인 막힌구멍이나 카운터보어와 같은 특징형상들은 평면도에 단일 엔티티로 구성되지만 포켓류의 형상들은 여러개의 엔티티들로 연결된 폐루프를

형성하고 있다. 포켓류 중 관통사각포켓과 막힌사각포켓 형상을 인식하기 위해서는 Fig. 7에서 보는 바와 같이 평면도내의 페루프를 구성하고 있는 line entity 1, 2, 3, 4의 존재여부를 확인 한다. 이 방법은 먼저 평면도의 기준점 b3 좌표값을 드로잉 데이터 베이스에서 AutoLISP내장함수인 entget과 assoc 함수를 사용하여 기준 좌표값을 추출한 후, b3 좌표의 x, y좌표값보다 큰 시작점과 끝점으로 이루어진 line entity를 모두 검색한다. 그러면 line entity 6, 7은 기준점 b3 좌표값과 동일한 한쪽 끝점의 좌표값을 가지므로 제외되고 5, 8의 line entity도 한쪽 끝점의 x, y좌표값이 기준점 b3의 min_x3, min_y3와 각각 일치하므로 제외되어 line entity 1, 2, 3, 4를 평면도내에서 추출하게 된다. 다음으로 다른 투상도에서 평면도내의 포켓 특징형상과 관련되어 있는 line entity가 존재하는지를 확인하여 해당 위치에 line entity L1, L2가 존재하면 사각포켓으로 인식한다. 관통사각포켓과 막힌사각포켓 형상의 인식구별은 막힌구멍과 관통구멍 인식구별과 마찬가지로 기준점 b1의 z좌표값 min_z1과 L1, L2의 한쪽 끝점의 z좌표값이 일치하면 관통 사각포켓으로 인식하고, L1, L2의 양끝점의 z좌표값이 기준점 b1의 z좌표값보다 크면 막힌 사각포켓으로 인식한다.

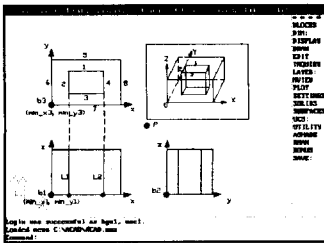


Fig. 7 3-view drawing and 3D solid object of closed through pocket

3.4.4 아크포켓 인식 규칙

아크포켓형상의 특징은 Fig. 8에서와 같이 평면도에 두 개의 마주보는 원호(Arc)가 존재하고 두 원호의 끝점을 연결하는 선으로 페루프를 이루고 있다. 이 특징형상을 인식하기 위해서는 드로잉 데이터 베이스에서 원호(Arc)에 대한 데이터를 추출한 다음 원호의 시작점과 끝점의 좌표 sp1, ep1, sp2, ep2의 값을 추출한다. 그런 다음 두 원호 2, 4을 연결하는 line 1, 3가 존재하여 원호 2, 4와 페루프를 이루는지를 검색후, 다른 투상도에서 아크포켓 형상과 관련된 line entity L1, L2가 존재하면 아크포켓으로 인식한다.

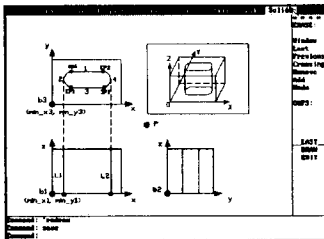


Fig. 8 3-view drawing and 3D solid object of arc-pocket

3.4.5 Pocket Partial Round 인식 규칙

Pocket partial round 형상의 특징은 아크포켓 형상과 달리 평면도내에 1개의 원호와 3개의 line entity로 이루어진 페루프를 구성하고 있다. 원호의 시작점과 끝점은 드로잉 데이터 베이스에서 추출한후, 원호의 양 끝점 sp2, ep2를 연결하는 line 1, 3이 존재하는지를 검색한다. line 1, 3의 존재여부 확인은 사각포켓 형상을 인식하는 방법과 유사한 방법으로 평면도의 기준점 b3 좌표값보다 큰 line entity를 모두 검색하여 페루프를 이루고 있는 line entity 들만 추출한다. 다음으로 검색된 line 1, 3의 양끝점을 연결하는 원호 이외의 선분 2가 존재하고, 정면도와 측면도에 이 형상과 관련된 해당위치에 line이 존재하면 pocket partial round로 인식한다. Pocket partial round이 포함되어 있는 투상도는 Fig. 9에 있다.

3.4.6 막힌슬롯 인식규칙

막힌슬롯 형상의 투상도는 Fig. 10에 보인 것과 같이 슬롯형상을 인식하기 위해서는 평면도내에 서로 연결된 3개의 line entity가 존재하고 3개의 line중 2개의 line entity의 시작점이나 끝점의 y좌표값이 평면도의 기준점 b3좌표의 y좌표값 min_y3과 같아야 한다. 평면도내의 슬롯형상을 나타내고 있는 line 1, 2, 3의 존재를 확인한 후에 다른 투상도에서 슬롯형상과 관련된 line entity가 존재하면 막힌슬롯으로 인식한다. 슬롯형상을 나타내는 line 1, 2, 3을 추출하기 위해서는, 먼저 드로잉 데이터 베이스에서 평면도의 기준점 b3 좌표값을 추출한 다음, 평면도내에 line entity를 검색하여 b3 좌표의 x좌표 min_x3와 동일한 x좌표값을 갖는 line entity를 제외시키면 line 1 ~ 7중에서 line 4, 5, 6의 한쪽 끝점은 b3좌표값과 일치하므로 제외되고 line 1, 2, 3, 7이 남게 된다. 막힌슬롯 형상을 이루기 위해서는 3개의 line entity로 연결되어 있어야 하므로 7번 line entity도 제외되어 line 1, 2, 3이 추출되어 막힌슬롯 형상을 이루게 된다.

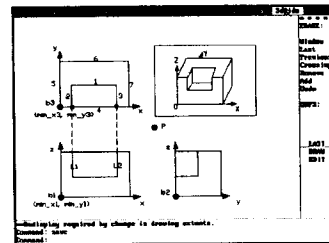


Fig. 10 3-view drawing and 3D solid object of blind slot

3.5 복합형상 인식 알고리즘

한 부품도내에서는 몇 개의 기본 특징형상들이 조합을 이루어 새로운 복합 형상을 구성하고 있고, 이들 기본형상들에 대한 계층적 구조(hierarchical structure) 관계가 형상인식의 순서와 가공 우선 순위를 결정하는데 중요한 지표로 사용되어 질 수 있다. 따라서 기본 특징형상들에 관한 상호

연관관계(topology)를 정립할 필요가 있다. 복합형상을 구성하고 있는 기본 특징형상들은 동일한 형상이라고 하더라도 상호 위치 관계에 따라 차등적 계층구조를 갖는다. 즉, 어떠한 형상가공에 의하여 생성된 형상가공면에 또다른 형상이 존재하면 후자는 전자의 child이고 전자는 후자의 parent로 정의한다. 그리고 동일한 parent의 child인 형상들끼리는 brother 관계로 정의한다.^[5] 특징형상 인식 순서는 인식된 결과 데이터를 이용한 입체 모델링과 가공 순위를 결정하는데 중요한 역할을 한다. 본 연구에서 수행된 형상인식의 특징은 설계단계에서 특징형상을 그리는 순서대로 형상인식을 수행하므로 parent-child 관계에 있는 복합형상은 가공시에 parent 형상을 먼저 가공하고 child형상은 뒤에 가공하므로 설계시에도 이와 마찬가지로 parent, child 순서로 그려준다. brother 관계에 있는 child가 여러개 있을 경우에는 가공 우선 순위를 정하여 특징형상들을 그려준다. 이렇게 하여 특징형상 정보를 인식되어진 순서대로 일정한 파일로 저장하여 3차원 형상모델링과 가공데이터로도 이용할 수 있게 한다. Fig. 11은 아크포켓 형상안에 두 개의 관통구멍이 들어 있는 parent-child관계와 brother관계를 이루고 있는 기본 특징형상들의 조합인 복합형상을 보여주고 있다. 형상인식의 순서는 설계단계에서 특징형상을 그리는 순서대로 인식을 하므로 parent관계에 있는 아크포켓 형상을 먼저 인식하고, hole1, hole2 순서로 형상인식을 수행한다.

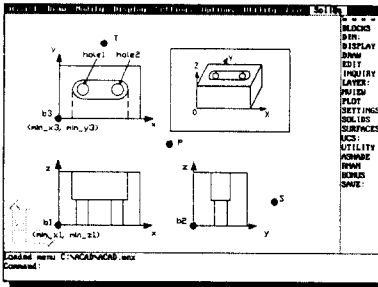


Fig. 11 3-view drawing and 3D solid object of compound feature

4. 특징형상 3차원 형상 모델링

모든 특징형상들이 인식되면 형상인식이 정확하게 수행되었는지 검증하고 가시화(visualization)하기 위하여 형상인식된 각 특징형상들의 인식 데이터 이용하여 3차원 솔리드 모델링을 수행한다. 이 작업은 ADS환경의 C언어와 AME의 API(Application Programming Interface)에서 제공되는 AME 기본 솔리드 생성함수, 규칙화된 불리언 연산(Boolean operators) 함수를 사용하여 작성된 프로그램을 통하여 3차원 기본 형상을 생성한다. 본 연구에서는 CSG(Constructive Solid Geometry) Representation을 트리(tree)구조로 저장하는 데이터 구조를 사용하여 불리언 연산작업인 합집합, 차집합, 교집합을 사용하여 모델링이 수행하였다. Fig. 12은 형상인식이 수행된 아크포켓 형상의 데이터 구조를 보여주고 있

고, Fig. 13은 인식된 데이터 파일 Fig. 12을 입력 데이터로 사용하여 최종 모델링 형상 $P = F - (A + B + C)$ 관계가 성립되는데 것을 보여주고 있는데 특징형상 모델링 순서는 다음과 같다.

- [step 1] 먼저 형상 인식된 결과 데이터를 입력 데이터로 사용하여 3 차원 기본형상 B(box)를 생성한다.
- [step 2] 다음으로 A(hole)형상을 인식되어진 치수대로 생성하여 B형상쪽으로 이동시키고 불리언 연산함수를 ap_assocmat를 사용하여 D(hole+box)형상을 완성한다.
- [step 3] C(hole)형상을 생성하여 D형상쪽으로 ap_translate 함수를 사용하여 이동시키고, 결합시켜 E형상인 아크포켓 형상을 만든다.
- [step 4] 생성된 3차원 기본형상(아크포켓)을 가공전 원재료에 해당되는 F(box)형상의 해당위치로 옮긴 후, 불리언 연산 함수 ap_diff함수를 사용하여 최종적인 형상 P형상인 완성되어 아크포켓의 3차원 모델링이 완성된다.

ap.dat	
OR	
x_or = 3.0	AP
y_or = 1.8	x_c = 3.0
z_or = 0.0	y_c = 3.0
	z_c = 4.0
BOX	L1 = 4.0
L = 10.0	W1 = 2.0
W = 6.0	dp = 2.0
H = 4.0	dir = -3

Fig. 12 Recognition result data structure of arc-pocket

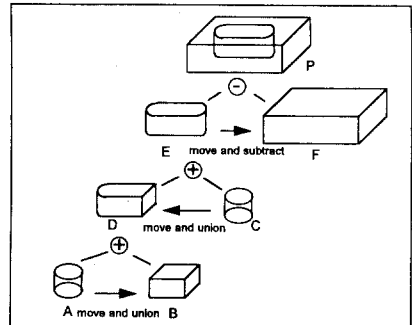


Fig. 13 3D feature modeling process of arc-pocket

Fig. 14는 아크포켓 형상에 대해 3차원 입체 형상 모델링하는 AME 프로그램의 일부를 보여 주고 있다. 본 연구에서 형상인식이 수행된 다른 기본 형상들도 이와 같은 방법으로 모델링된다.

```

ap_box (Lx, Ly, Lz, &bo[0]) ;
ap_box (sx, sy, sz, &ap[1]) ;
ap_cylinder(rx, ry, h, &cy[0]) ;
ap_translate(x1, y1, z1, ap_mat1) ;
ap_assocmat(ap[1], ap_mat1) ;
ap_cylinder(rx, ry, h, &cy[1]) ;
ap_translate(x2, y2, z2, ap_mat2) ;
ap_assocmat(ap[1], ap_mat2) ;
arp[2] = (ap_solid) NULL ;
ap_union (arp, &ap[1]) ;
ap_translate(x3, y3, z3, arp) ;
ap_diff(bo, arp, &sol) ;
ap_postsol(sol, AP_POSTWIRE) ;

```

Fig.14 Example of automatic AME programming in arc-pocket

5. 적용 사례

본 연구를 통하여 개발된 특징형상 인식시스템은 기본 특징형상인 구멍류(관통구멍, 막힌구멍, 카운터보어, 카운터싱크), 포켓류(관통사각포켓, 막힌사각포켓, 아크포켓, Pocket partial round), 슬롯류 (막힌슬롯, Slot partial round) 등의 형상과 기본 특징형상들의 조합인 복합형상 등 프레스 금형 도면에서 가장 많이 사용되는 특징형상들에 대해서 적용하였다.

Fig. 15(a)는 기본 특징형상들로만 구성되어 있는 프레스 금형 도면을 특징형상 인식시스템의 입력 도면으로 사용되는 것을 보여주고 있고, 입력된 도면으로부터 특징형상 인식 규칙에 따라 형상인식을 수행하는 과정과 결과 데이터를 모니터에 display하는 것을 Fig. 15(b)에서 보여 준다. 또한 인식된 결과 데이터를 이용한 가공데이터와 3차원 특징형상 모델링을 수행할 수 있는 데이터 구조로 파일에 저장하는 예는 Fig. 15(c)과 같다. Fig. 15(d)은 Fig. 15(c)의 데이터 파일을 사용하여 ADS와 AME를 이용하여 3차원 모델링 과정을 거쳐 와이어프레임(wire frame)형태로 모델링한 것이다. Fig. 15(e)는 와이어 프레임 형태에서 Solmesh, Hide, shade 명령을 사용하여 대상물 내부가 채워진 mesh block형태로 변형시켜 은선을 제거하고 명암처리를 하여 모델링한 것이다.

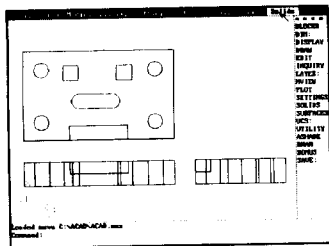


Fig. 15(a) Example drawing : 3-view drawing

```

*****
* MAIN MENU *
*****
* 1. FEATURE RECOGNITION *
* 2. 3-D MODELING *
* 3. RECOGNITION AND MODELING *
* 4. QUIT *
*****

Choose the Number : 1
blipmode ON/OFF <Off> : off
Command :
Input the base point :
Input the side view point :
Input the plane view point :
(2.0 1.0)
(120.0 1.0)
(2.0 16.0)
Enter the data file name : Featl.dat

HEXAHEDRON
POINT-1 = (0 0 0)
POINT-2 = (100.0 0 0)
POINT-3 = (0 60.0 0)
POINT-4 = (0 0 10.0)
LENGTH(x) = 100.0
WIDTH(y) = 60.0
HEIGHT(z) = 10.0

HOLE-1 (Through)
Center point : (12.0 12.0 10.0)
Direction : -3
    
```

Fig. 15(b) Display feature recognition data

Featl.dat

```

OR
x_or = 3.0
y_or = 1.8
z_or = 0.0

BOX
L = 100.0
W = 60.0
H = 10.0

THO
x_c = 12.0
y_c = 12.0
    
```

Fig. 15(c) Recognized feature data structure for 3D feature modeling

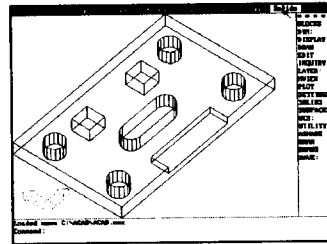


Fig. 15(d) Example drawing : Wireframe modeling

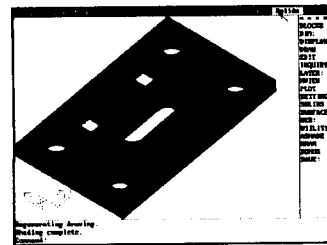


Fig. 15(e) Example drawing : Solid modeling using shading

6. 결론

본 연구에서는 CAD시스템으로부터 가공 특징형상을 자동으로 인식하고 추출하는 특징형상 인식 시스템을 개발하는데 목적을 두었으며, 대상부품으로서 비회전형상을 대상으로 프레스 금형부품의 드릴링 밀링등 절삭가공작업에 의하여 생성될수 있는 특징형상을 고려하여 형상인식을 수행하였고, 여기서 나온 인식데이터를 이용하여 3차원솔리드 입력데이터와 가공데이터로 활용될수 있도록 하였다.

참고 문헌

- [1] Erve, A.H. Van't, and Kals, H.J.J., "The selection of optimum Maching Operations in Automated Process Planning", Proc. of 19th CIRP International Seminar on Manufacturing System, pp.47-54, June, 1987.
- [2] Wang, H.P., and Wysk, R.A., "AIMSI : A Prelude to a New Generation of Integrated CAD/CAM System," INT. J. Vol,26, No1, pp.119-131, 1988.
- [3] Shah, J.J., "Philosophical Development of form Feature Concept", in Processing of the CAM-I Features Symposium, PP-90-PM-02, pp.55-77., 1990.
- [4] Joshi, S., "CAD Interface for Automated Process Planning," Ph.D. Thesis, Purdue University, Aug. 1987.
- [5] Esakov, J., and Weiss, T., " Data structure : An Advanced Approach Using C", Printice-Hall. 1989