

A novel feature recognition solution for Design & Process Planning integration based on IGES data structure

Mahmoud Houshmand, Omid Fatahi Valilai

Department of Industrial Engineering, Sharif University of
Technology, Tehran, Iran
Hoshmand@sharif.edu, FValilai@sharif.edu

Reza Ghasemi

Department of Industrial Engineering, Mazandaran
University of Science and Technology, Mazandaran, Iran
Reza.jourghasemi@b-just.ac.ir

Abstract— In recent years, many researchers have proposed different solutions and frameworks to integrate CAD and CAPP concepts. Of the most remarkable solutions are the automatic feature recognition tools. These tools work with CAD systems and highly impact the level of integration. CAD files contain detailed geometric information of the parts, which are not suitable for using in the downstream applications such as process planning. To overcome this shortcoming, this paper proposes an intelligent feature recognition solution to enable a feature recognition system. This system has the ability to communicate with various CAD/CAPP systems. The proposed solution is developed for 3D prismatic parts that are created by using solid modeling package using B-Rep technique—a famous geometric modeling technique—as a drawing tool. The system takes a neutral file in Initial Graphics Exchange Specification (IGES) format as input and interprets the information to process planning and manufacturing information. The B-Rep geometrical information of the part design is then analyzed by a feature recognition program which will extract the manufacturing features from the geometrical information. The feature recognition module uses a novel idea based on a geometric reasoning approach and it is implemented by using object oriented design software package. Moreover, the feature recognition algorithm can recognize a wide variety of manufacturing features such as step, holes, etc. The further research to improve the proposed solution for compatibility with ISO 10303 standard is highly recommended.

Keywords—Computer Integrated Manufacturing (CIM); CAD; CAPP; Integration, IGES data format

I. INTRODUCTION

Today's competitive environment of mechanical manufacturing enterprises necessitates the digital manufacturing and computer integrated manufacturing (CIM) Paradigm [1]. Integration of Computer Aided Design (CAD) and Computer Aided Process Planning (CAPP) has been known to play a key role in achieving digital manufacturing and computer integrated manufacturing paradigm. Moreover, researchers believe that this integration is vital to enable capabilities for responding rapidly to market changes [1,2]. CAD & CAPP systems can help manufacturing enterprises to reduce design and planning time and also increase consistency and efficiency [2]. However, the main integration challenge is investigated to be the CAD data transferring mechanisms to a downstream Computer Aided Manufacturing (CAM) system, in order to fulfil the CIM paradigm. This challenge is due to

the lack of neutral formats as well as content to convey the CAD information [3-4].

The integration of CAD and CAM has received significant attention through recent years according to the development of Information Technology capabilities. However, the actual solutions for integration between CAD and CAM, for the downstream applications such as process planning, can be achieved only when the manufacturing information can be obtained directly from 3D solid model [5-7]. To enable the collaboration among various CAD & CAPP software packages in a CIM system, one of the fundamental solutions is to extract and identify the information in the CAD model file [8-10]. The conventional approach was to conduct feature extraction by the human planner through examining the part and recognizing the features designed into the part. In the CIM paradigm, automated feature recognition can best be applied by CAD systems capable of generating the product geometry based on features, thereby making it possible to capture information about tolerance, surface finish, etc. [11]. However, such CAD systems are not yet mature in feature recognition and their wide applications in different related domains cannot be accomplished yet [43]. It is therefore necessary to propose intelligent feature recognition solutions to extract features from part geometry.

In this paper, a novel solution for design feature analysis and extraction of prismatic parts is proposed. This solution aims to achieve the integration between CAD and CAPP. This solution consists algorithms which can be applied by the manufacturing engineers to increase their productivity in process planning, and selecting manufacturing machines and tools. This solution is based on Initial Graphics Exchange Specification (IGES) data format. This data format is used to enable a data model of manufacturing features in the proposed solution. As the IGES is a known and widely used data format, it insures the application of proposed solution for integration and sharing of CAD data.

II. RELATED RESEARCHES AND STUDIES

A. CAD & CAPP

The conventional partition of design and manufacturing activities in modern industry is broadly recognized as a key

contributor to product development costs [11]. Traditional methods of design usually use low level methods of geometric representation like wireframe representation, boundary representation (B-Rep), and Constructive Solid Geometry (CSG) representation [12]. Information needed for higher level applications, such as process planning, could not be extracted from these methods. One of the significant solutions to overcome these shortcomings was the feature based design and manufacturing. The word “feature” signifies different meaning in different contexts depending on the specific domain. For example, in design it refers to a web or a notch section, etc., while in manufacturing it refers to slots, holes, and pockets, while in inspection it is used as a datum or reference on a part [17]. Feature based modeling has proven to be an effective and time saving approach for product design. Features give a way to handle the design at a higher level of description than engineering drawings or traditional CAD descriptions that characterize a designed part in terms of mathematical surfaces or volumes [12-16].

Various researches proposed solutions and algorithms for feature based design and process planning. Of these dominant researches are syntactic pattern recognition [18-19], volume decomposition [20-21], expert system and logic [22-23], the CSG-based approach [24], the graph-based approach [25], and the neural-network-based approach [26-27]. Zhao, *et al.* [28] used a wireframe model to develop a methodology and algorithm of recognizing machined features by using graph theory. Two dimension representation was used to generate closed boundaries of the machined parts from wireframe models. The method was powerful especially for symmetrical work pieces such as canonical parts in which all vertices have degree of three. The elimination of the vertices with degree greater than three required much more computations due to the difficulties of determining each edge of the facets. This methodology did not use 3D solid modeling and any kind of standard formats for representing the designed part.

Rozenfeld and Kerry [29] tried to present solution for automated process planning for parametric parts, which at the same time fulfill the process planning requirements of any other parts and products in a real industrial environment. The parts were presented in their CAPP system by parametric CAD module. In this representation, the parts were represented by parameters, such as the ones described through features and their attributes (a hole with its diameter, length, tolerance, and roughness). However, the types of features which CAPP system could applied were nor defined.

Lee, *et al.* [30] addressed a progressive solid model generation. To reduce the complexity of assembly models, researchers [31-32] proposed a multi-resolution decomposition of an initial B-Rep assembly model. Lockett [33] proposed to recognize specific positive injection molding features. The proposed method used an already generated Medial Axis (MA) to find features from idealized models. One recognized difficulty was the ability to obtain main a wide range of configurations. Shapiro [34] addressed B-Rep to CSG conversion as a means to associate a construction tree with a B-Rep model and Buchele [35] moreover applied it to reverse engineering configurations. Li, *et al.* [36-37] introduced a regularity feature tree used to highlight the symmetry in the

object that differs from CSG and construction trees. Belaziz, *et al.* [38] proposed a morphological analysis of solid models based on features and B-Rep transformations that were able to simplify the shape of an object. It was also a type of B-Rep to CSG conversion. Indeed, the shape modifiers were elementary B-Rep operators that did not convey shape information.

B. Feature Recognition Information model

IGES is a standard format which has been used to define the data related to object drawings in solid modeling CAD systems in B-Rep structure. Object’s geometric and topological information in IGES format can be represented by the entry fields that constitute the IGES file [39-40]. The geometric information are in the form of low level entities such as lines, planes, circles, and other geometric entities for a given object (designed part), and the topological information that defines the relationships between the object’s geometric parts are represented.

Similar to the most CAD systems, IGES is based on the concept of entities. Entities could range from simple geometric objects, such as points, lines, plane, and arcs, to more sophisticated entities, such as subfigures and dimensions. Entities in IGES are divided into three categories [41]:

1. Geometric entities such as arcs, lines, and points that define the object

2. Annotation entities such as dimensions and notes that aid in the documentation and visualization of the object

3. Structure entities that define the associations between other entities in IGES file.

Syntactically, IGES descends from a fixed format ‘card image’. Each input line (no longer called a card) is therefore limited to 80 characters, of which the last eight characters form a sequence number. Each line also has a one-character section character (‘S’, ‘G’, ‘D’, ‘P’ and ‘T’). The IGES standard defines about 80 entities; however, many of these have ‘forms’ that expand the total to about 150 entities [42]. An IGES file consists of five sections which must appear in the following order: Start section. Global section, Directory Entry (DE) section. Parameter Data (PD) section, and Terminate section, (as shown in Fig. 1).

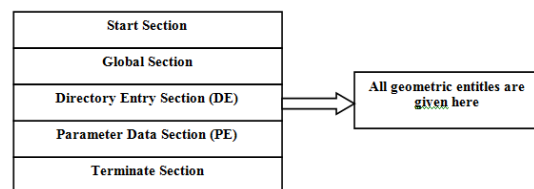


Fig. 1. IGES file structure

Fig. 2 shows a small IGES file. This file contains only two points (Type 116), two circular arcs (Type 100), and two lines (Type 110) entities. It represents a slot, with the points at the centers of the two half-circles that form the ends of the slot, and the two lines that form the sides.

III. PROPOSED FEATURE RECOGNITION SOLUTION FOR CAD & CAPP INTEGRATION

A. Assumptions

In this paper, the part design is introduced through CAD software and it is represented as a solid model by using Boundary Representation (B-Rep) technique as a design tool. The solid model of the part design consists of small and different solid primitives combined together to form the required part design. The CAD software generates and provides the geometrical information of the part design in the form of an ASCII file (IGES) format that is used as standard format. IGES provides the communication procedure for the proposed solution with the various CAD/CAM systems.

									S	1
1H,,1H,,4HSL0T,37H91SDUA2:[IGESLIB.BDRAFT.B2I]SLOT.IGS,,									G	1
17HBravo3 BravoDRAFT,31HBravo3->IGES V3.002 (02-Oct-87),32,38,6,38,15,									G	2
4HSL0T,1.,1.,4HINCH,8,0.08,13H871006.192927,1.E-06,6.,									G	3
31HD. A. Harrod, Tel. 313/995-6333,24HAPPLICON - Ann Arbor, MI,4,0;									G	4
116 1 0 1 0 0 0 0 0 0	1D	1								
116 1 5 1 0 0 0 0 0 0	0D	2								
116 2 0 1 0 0 0 0 0 0	1D	3								
116 1 5 1 0 0 0 0 0 0	0D	4								
100 3 0 1 0 0 0 0 0 0	1D	5								
100 1 2 1 0 0 0 0 0 0	0D	6								
100 4 0 1 0 0 0 0 0 0	1D	7								
100 1 2 1 0 0 0 0 0 0	0D	8								
110 5 0 1 0 0 0 0 0 0	1D	9								
110 1 3 1 0 0 0 0 0 0	0D	10								
110 6 0 1 0 0 0 0 0 0	1D	11								
110 1 3 1 0 0 0 0 0 0	0D	12								
116,0.,0.,0.,0,0,0;	1P	1								
116,5.,0.,0.,0,0,0;	3P	2								
100,0.,0.,0.,0.,1.,0.,-1.,0,0;	5P	3								
100,0.,5.,0.,5.,-1.,5.,1.,0,0;	7P	4								
110,0.,-1.,0.,5.,-1.,0.,0,0;	9P	5								
110,0.,1.,0.,5.,1.,0.,0,0;	11P	6								
S	1G	4D	12P	6					T	1

Fig. 2. IGES sample file [42]

B. Solution framework and structure

The proposed solution consists of a feature recognition program which analyses the B-Rep geometrical information of the part design. This program then extracts the features from the geometrical information based on geometric reasoning approach. This program is an object oriented design software which is built through C# language. The feature recognition program is able to recognize following features:

- Slot (through)
- Hole (through)
- Step (through)

These features can lead to required manufacturing information which are mapped through process planning phase. Fig. 3 shows the structure of the proposed solution. The proposed solution consists of three main components:

- A data file converter
- An object form feature classifier and
- A manufacturing features classifier

The first component converts a CAD data in IGES format into a proposed object oriented data structure. The second

component classifies the geometric features of the designed part obtained from the data file converter into different feature groups. The third component maps the extracted features to process planning point of view.

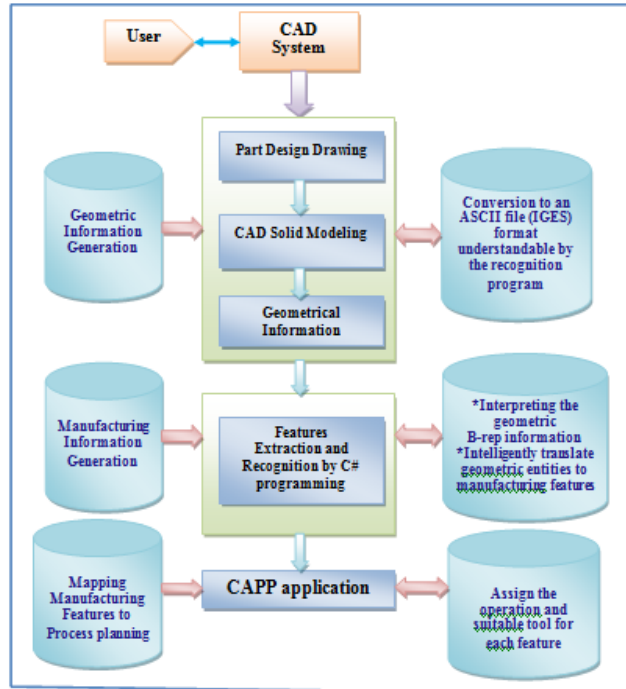


Fig. 3. Structure of the proposed solution

```
private void Form1_Load(object sender, EventArgs e)
{
    List<Color_definition> Color_def_;
    List<Transformation_matrix> transformationMatrix_;
    List<Property> Property_;
    List<Straight_line> Straight_line_;
    List<Rotational_B_spline_curve> Rotational_B_spline_curve_;
    List<surface_of_revolution> surface_of_revolution_;
    List<Composite_Curve> Composite_curve_;
    List<Curve_on_parametric_surface> Curve_on_parametric_surface_;
    List<trimmed_parametric_surface> trimmed_parametric_surface_;
    List<Plane> Plane_;
    List<Circle_arc> Circle_arc_;

    private void button1_Click(object sender, EventArgs e)
    {
        Color_def_ = new List<Color_definition>();
        transformationMatrix_ = new List<Transformation_matrix>();
        Property_ = new List<Property>();
        Straight_line_ = new List<Straight_line>();
        Rotational_B_spline_curve_ = new List<Rotational_B_spline_curve>();
        surface_of_revolution_ = new List<surface_of_revolution>();
        Composite_curve_ = new List<Composite_Curve>();
        Curve_on_parametric_surface_ = new List<Curve_on_parametric_surface>();
        trimmed_parametric_surface_ = new List<trimmed_parametric_surface>();
        Plane_ = new List<Plane>();
        Circle_arc_ = new List<Circle_arc>();
        string filename = @"F:\Part11.igs";
    }
}
```

Fig. 4. Structure developed codes of IGES lines

In the first component, all of data lines in IGES format are called by the developed module which is shown in Fig. 4 and then all of entities in directory section and parameter section in IGES format are called respectively. Points, vertices, lines, faces are extracted in the developed module as well as shown

in Fig. 5. In the second phase, when all entities consisting of all points, vertices, lines and faces are extracted, the program starts to run its features recognition module.

```
public double X1,X2,X3,Y1,Y2,Y3,Z1,Z2,Z3;
public Straight_line(string line)
{
    // TODO: Complete member initialization
    this.line = line;
    int index = line.LastIndexOf('D');
    if (index >= 0)
    {
        string s = line.Substring(index + 1);
        while (line[0] == ' ')
            s = s.Substring(1);
        key_ = Convert.ToInt32(s);
    }
}
internal void LoadProperty(string line)
{
    int index = line.IndexOf(',');
    line = line.Substring(index + 1);
    index = line.IndexOf(',');
    string s = line.Substring(0, index);
    X1 = ToDouble(s);
    line = line.Substring(index + 1);
    index = line.IndexOf(',');
    s = line.Substring(0, index);
    Y1 = ToDouble(s);
    line = line.Substring(index + 1);
}
```

Fig. 5. Structure of line entity in developed module

At the end of the phases, the solution has retrieved all circles, straight lines and composite curves from the CAD data. Circles, composite curves and straight lines are entities which form the design features of the part. All the entities are then sent to the main module with their subsets which consist of all points, lines, vertices and faces as well as shown in Fig. 6.

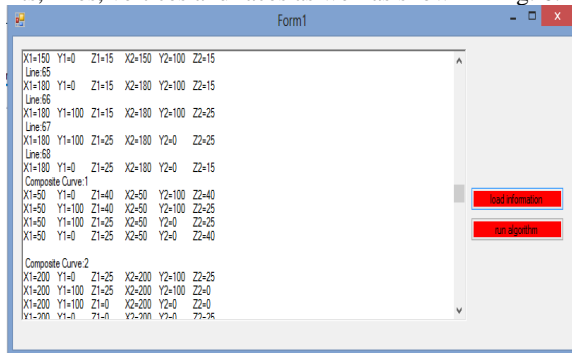


Fig. 6. The information of the lines, composite curves, circles with their related points, vertices and faces

Proceeding, all lines, composite curves and circle are sent to the top module with their subsets that including start points, end points, vertices and faces. Then the solution starts to run its feature recognition algorithm. In this paper, a novel idea in feature recognition is fulfilled through a specific mechanism that starts to search in the entities and finds all candidate features in the part design. For example, if more than 4 lines which make a loop are recognized then there will be a probability of a feature. The header structure of the mechanism is shown in Fig. 7. After the mechanism has finished the algorithm execution, all recognized features are presented as shown in Fig. 8. The proposed solution in this paper is able to extract a wide variety of applicable manufacturing features. Each feature has its own algorithm like:

Feature: STEP THROUGH (shown in Fig. 9)

For every inner edge (e1) of type line in the edge list

If face f2 is a adjacent to face f1

If two common faces(face1 and face 2) of the edge (e1) orthogonal to each other

And the angle between f2 and f1 is 90

Then faces f1 and f2 form a step

Create a new step T object and add to feature list

End for

```
class Slot
{
    public int index;
    public vector SL1_v, SL2_v, SL3_v;
    public process_planner_optimizer.Composite_Curve CC;
    public process_planner_optimizer.Straight_line SL1;
    public process_planner_optimizer.Straight_line SL2;
    public process_planner_optimizer.Straight_line SL3;
    public string ToString()
    {
        string s = "Slot: from component number " + (index+1) + "\n";
        s += SL1.PrintINFO() + "\n" + SL2.PrintINFO() + "\n" + SL3.PrintINFO() + "\n";
        return s;
    }
}

internal bool Z_IsSame(Slot s)
{
    return s.SL1.Z1 == SL1.Z1 && s.SL1.Z2 == SL1.Z2 && s.SL1.Z3 == SL1.Z3;
}
```

Fig. 7. Header structure of feature recognition in developed module

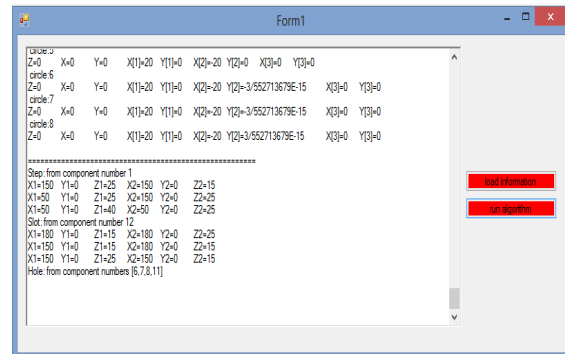


Fig. 8. The extracted feature representations

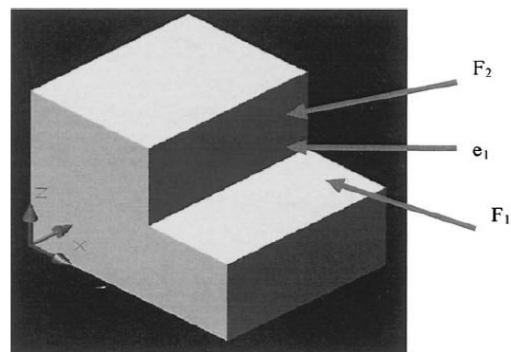


Fig. 9. The STEP THROUGH feature

Feature SLOT THROUGH (Fig. 10)

For every two inner edges of type line in the edge list.
If face f2 adjacent to face f1 and face f3 is adjacent to face f1

If the two edges have a common face(face1) connected to them and inner edge count of the outer loop of the face equals two.

And face f2 and face f3 are perpendicular to the common face(face1) and parallel to each other with inner edge count of their outer loops equal one.

The faces f1, f2 and f3 form a slot.

Create a new slot T object and add to feature list.

End for

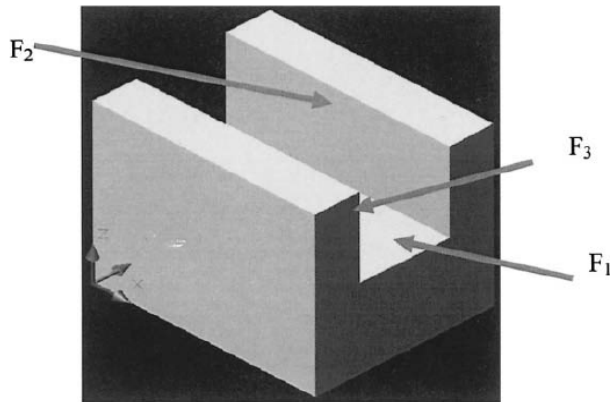


Fig. 10. The SLOT THROUGH feature

Feature HOLE THROUGH (Fig. 11)

For every edge in the list, If edge is a closed loop and is inner

Then create a new HOLE THROUGH object with the edge as the circle surface and look for the similar edge in the list

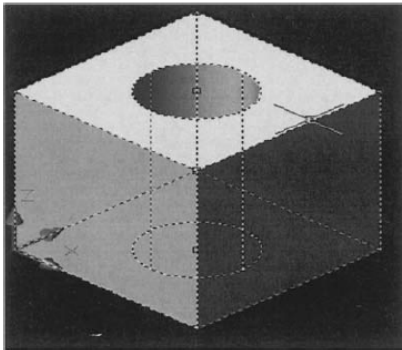


Fig. 11. The HOLE THROUGH feature

IV. CONCLUSION

Today's competitive environment of mechanical manufacturing enterprises necessitates the digital manufacturing and computer integrated manufacturing (CIM)

Paradigm In recent years, many researchers have proposed different solutions and frameworks to integrate CAD and CAPP domains. Of the most remarkable solutions are the automatic feature recognition tools. The feature recognition tools play an important role in connecting the bridge between CAD and CAPP systems and highly impact the level of integration. Despite of the proposed researches for feature recognition, there is a need for efficient solutions especially considering the variety of CAD geometric files. CAD files contain detailed geometric information of the parts, which are not suitable for using in the downstream applications such as process planning. To overcome this shortcoming, this paper proposed an intelligent feature recognition solution to enable an efficient feature recognition system. This system has the ability to communicate with various CAD/CAPP systems. The proposed solution is developed for 3D prismatic parts that are created using B-Rep technique-a famous geometric modeling technique-as a drawing tool. The system takes a neutral file in Initial Graphics Exchange Specification (IGES) format as input and interprets the information to process planning and manufacturing information. The B-Rep geometrical information of the part design is then analyzed by a novel proposed feature recognition algorithm which extracts the manufacturing features from the geometrical information. The feature recognition module uses a novel idea based on a geometric reasoning approach and it is implemented using object oriented design software package. Moreover, the feature recognition algorithm is able to recognize a wide variety of manufacturing features such as steps, slots, holes, etc. The further research to improve the proposed solution for compatibility with ISO 10303 (STEP) standard is highly recommended. Moreover, the integration of the proposed solution with CNC machining domain is also recommended.

V. REFERENCES

- [1] B. Khoshnevis, D. Sormaz, JY. Park. An integrated process planning system using feature reasoning and space search-based optimization. *IIE Transactions* 1999;31:597-616.
- [2] H. Nagaraj, S., and B. Gurumoorthy. "Machinable volume extraction for automatic process planning." *IIE Transactions* 34.4 (2002): 393-410.
- [3] N. Ahmad, & A. Haque. Manufacturing feature recognition of parts using DXF files. Fourth International conference on Mechanical Engineering, Dhaka, Bangladesh, (2001): (Vol. 1, pp. 111-115).
- [4] D. Natekar, X. Zhang, G. Subbarayan. Constructive solid analysis: a hierarchical, geometry-based meshless analysis procedure for integrated design and analysis. *Computer Aided Design*, (2004): 36(5), 473-486.
- [5] H-C. Chang, WF. Lu, FX. Liu. Machining process planning of prismatic parts using case-based reasoning and past process knowledge. *Applied Artificial Intelligence*, (2002): 16(4), 303-331.
- [6] S. Mansour. Automatic generation of part programs for milling sculptured surfaces. *Journal of Materials Processing Technology*, (2002): 127(1), 31-39.
- [7] HK. Miao, N. Sridharan, J. Shah. CAD/CAM integration using machining features. *International Journal of Computer Integrated Manufacturing*, (2002): 15(4), 296-318.
- [8] MP. Groover. *Automation, Production Systems, and Computer-Integrated Manufacturing*. New Jersey, USA: Prentice-Hall, Inc. (2001)..
- [9] V. Han, I. Han. Manufacturable feature recognition and its integration with process planning. *Proceedings of the Symposium on Solid Modeling and Applications*, (1999): 108-118.

- [10] L. Roucoules, O. Salomons, H. Paris. Process planning as an integration of knowledge in the detailed design phase. *International Journal of Computer Integrated Manufacturing*, (2003): 16(1), 25–37.
- [11] MW. Fu, WF. Lu, SK. Ong, IBH. Lee, AYC. Nee. An approach to identify design and manufacturing features from a data exchanged part model. *Computer Aided Design*, (2003): 35(11), 979–993.
- [12] A. Grayer. Geometric modeling in production. *SME Technical Paper*, (1979): 79 (150), 15p.
- [13] K. Lee. *Principles of CAD/Cam/CAE systems*. Addison Wesley. (1999).
- [14] S. Linardakis, AR. Mileham. Manufacturing feature identification for prismatic components from CAD DXF files. *Advances in Manufacturing Technology VII*. In Proceedings of 9th National Conference on Manufacturing Research, (1993): pp. 37–41.
- [15] CH. Liu, DB. Perng, Z. Chen. Automatic form feature recognition and 3D part recognition from 2D CAD data. *Computer and Industrial Engineering*, (1994): 26(4), 689–707.
- [16] TC. Woo. Visibility maps and spherical algorithms. *Computer Aided Design*, (1994): 26(1), 6–16.
- [17] CR. Devireddy, K. Ghosh. Feature-based modeling and neural networks-based CAPP for integrated manufacturing. *International Journal of computer Integrated Manufacturing*, (1999): 12(1), 61–74.
- [18] PC. Sreevalsan, JJ. Shah. Unification of form feature definition methods. Proceedings of the IFIP WG 5.2 Working Conference on Intelligent Computer Aided Design, (1992): 83–106.
- [19] SM. Staley, MR. Henderson, DC. Anderson. Using syntactic pattern recognition to extract feature information from a solid geometric database. *Computers in Mechanical Engineering*, (1983): 2(2), 61–66.
- [20] AC. Lin, SY. Lin. Volume decomposition approach to process planning for prismatic parts with depression and protrusion design features. *International Journal of Computer Integrated Manufacturing*, (1998): 11(6), 548–563.
- [21] HS. Nagaraj, B. Gurumoorthy. Machinable volume extraction for automatic process planning. *IIE Transactions*, (2002): 34(4), 393–410.
- [22] SS. Madurai, L. Lin. Rule-based automatic part feature extraction and recognition from CAD data. *Computers and Industrial Engineering*, (1992): 22(1), 49–62.
- [23] A. Munns., Y. Li, XC. Wang. A rule-based feature extraction from csg representations and an application in construction. Proceedings of SPIE-The International society of Optical Engineering, (1995): 2620(1), 269–276.
- [24] D. Natekar, X. Zhang, G. Subbarayan. Constructive solid analysis: a hierarchal, geometry-based mesh less analysis procedure for integrated design and analysis. *Computer Aided Design*, (2004): 36(5), 473–486.
- [25] S. Joshi, TC. Chang. Graph-based heuristics for recognition of machined features from 3D solid model. *Computer Aided Design*, (1988): 20(2), 58–66.
- [26] P. Chang, C. Chang. An integrated artificial intelligent computer aided process planning system. *International Journal of Computer Integrated Manufacturing*, (2000): 13(6), 483–497.
- [27] CR. Devireddy, K. Ghosh. Feature-based modeling and neural networks-based CAPP for integrated manufacturing. *International Journal of computer Integrated Manufacturing*, (1999): 12(1), 61–74.
- [28] Z. Zhao, SK. Ghos, D. Link. Recognition of machined surfaces for manufacturing based on wireframe models. *Journal of Materials Processing Technology*, (1990): 24(1), 137–145.
- [29] H. Rozenfeld, HT. Kerry. Automated process planning for parametric parts. *International Journal of Production Research*, (1999): 37(17), 3981–3993.
- [30] JY. Lee, JH. Lee, H. Kim, HS. Kim. A cellular topology-based approach to generating progressive solid models from feature-centric models. *Computer Aided Design* (2004); 36(3):217–29.
- [31] S. Kim, K. Lee, T. Hong, M. Kim, M. Jung, Y. Song. An integrated approach to realize multi-resolution of B-Rep model. in: Proceedings of the 2005 ACM symposium on solid and physical modeling. (2005): p. 153–62.
- [32] J. Seo, Y. Song, S. Kim, K. Lee, Y. Choi, S. Chae. Wrap-around operation for multiresolution CAD model. *Computer Aided Design Application* (2005); 2(1–4):67–76.
- [33] HL. Lockett, MD. Guenov. Graph-based feature recognition for injection molding based on a mid-surface approach. *Computer Aided Design*, (2005); 37(2):251–62.
- [34] V. Shapiro, DL. Vossler. Separation for boundary to CSG conversion. *ACM Trans Graph*, (1993); 12(1):35–55.
- [35] SF. Buchele, RH. Crawford. Three-dimensional half space constructive solid geometry tree construction from implicit boundary representations. *Computer Aided Design*, (2004); 36:1063–73.
- [36] M. Li, FC. Langbein, RR. Martin. Constructing regularity feature trees for solid models. in: Proc. geometric modeling and processing; LNCS, (2006): p. 267–86.
- [37] M. Li, F. Langbein, R. Martin. Detecting design intent in approximate CAD models using symmetry. *Computer Aided Design* (2010); 42:183–201.
- [38] M. Belaziz, A. Bouras, J-M. Brun. Morphological analysis for product design. *Computer Aided Design* (2000); 32(5–6):377–88.
- [39] M. Kahrs. The heart of IGES. *Software-Practice and Experience*, (1995). 25(8), 935–946.
- [40] K. Mark. The Heart of IGES. *Software-Practice and Experience*, (1995). 25(8), 935–946.
- [41] B. William, Initial Graphics Exchange Specification IGES 5.3, ANS US PRO/IPO-100 (1996).
- [42] MISC-IGES Example file Slot.
- [43] Valilai, O.F., Houshmand, M., LAYMOD; A layered and modular platform for CAx product data integration based on the modular architecture of the STEP standard, *International Journal of Computer Integrated Manufacturing*, (2012), 25, 473-487, <http://doi.org/10.1080/0951192X.2011.646308>