# Attribute Adjacency Matrix Approach for Extracting and Recognizing Manufacturing Prismatic Features from CAD Models

**Dr. Hussein. S. Ketan*  &  Omar. H. Hussoon***

Received on: 30/2/2008
Accepted on: 5/3/2009

**Abstract**

This paper introduces Developed Attributes Adjacency Matrix (DAAM) approach for extracting and recognizing of part's features, by deal with decoupling or isolated manufacturing features. The proposed algorithm consists of two stages, pre-processor stage which perform extraction of information from Design Exchange Format (DXF) file format in Computer Aided Design (CAD), and post-processor stage which recognizes part's features depending on attributes adjacency matrix.

The main contributions of this research include building the adjacent relations between faces of the part by geometrical characteristics in form of matrix, and use clustering operation to extract the exact faces of feature and use different classes of geometrical characteristics to assisting to recognize depression prismatic features useful in machining and other application such as dimensional inspection and assembly.

**Keyword:** Attribute Adjacency Matrix, Isolated feature, Feature extraction and recognition, CAD, DXF.

## طريقة مصفوفة تجاور الخواص لاستخراج وتمييز السمات التصنيعية المضلعة من نماذج التصميم المعان بالحاسوب

**الخلاصة**

يهدف البحث الى تطوير مصفوفة تجاور الخواص والخاصة باستخراج وتمييز السمات التصنيعية الغير مترابطة او المنعزلة المضلعة. وتم في العمل اقتراح خوارزمية تتكون من مرحلتين, المرحلة الاولى تتضمن المعالجة الاولية لبيانات الملف القياسي (DXF) والخاصة بالبيانات التصميمة للسمات والتي تم تصميمها باستخدام نظام التصميم المعان بالحاسوب (CAD) لغرض استخراج المعلومات الخاصة بالسمة,اما المرحلة الثانية تتضمن تمييز سمات الجزء بالاعتماد على صفات التجاور لخواص السمات. ان المساهمات الرئيسية في هذا البحث تتضمن بناء علاقات التجاور بين وجوه الجزء الهندسي على شكل مصفوفة, ثم استخدام عملية المجاميع (clustring) لاستخراج الوجوه الخاصة بالسمة واستعمال خواص هندسية مختلفة لتمييز السمات المضلعة (prismatic) المقعرة ذات الاوجه المتعددة لاغراض التطبيق في التشغيل والتطبيقات الاخرى مثل فحص الابعاد و التجميع.

*Production Engineering and Metallurgy Department, University of Technology/Baghdad


2085

## 1. Introduction

The gap in information between CAD and computer aided process planning (CAPP) or automated inspection planning is reflect by the fact these applications require as input form of features, where traditional CAD representation can provide only faces, edges and vertices[1,2]. Therefore, to overcome this deficiency in integration, several different mechanisms have been proposed to automatically extract the high-level representation called features from low-level entities in CAD representation. The ambiguity of feature representation is one of the major sources of difficulty for feature extraction mechanisms [3]. Various approaches have been developed for automatic feature recognition mechanisms that can be informally classified into the categories [4,5]: Syntactic pattern recognition approach, Graph-based approach, Rule-based approach, Volumetric approaches (including "cell- based" techniques) and Evidence-based reasoning approaches. Syntactic pattern recognition characterizes the overall part shapes as the composition of certain geometric primitives. Feature recognition proceeds by parsing the input syntactic expression of a part using grammar rules to identify the syntactic patterns representing the geometric primitives. In graph-based feature recognition, a graph is used to describe the topological shape of a part as well as that of a primitive feature. Feature recognition is performed by searching graph representation of the part to identify a sub-graph that matches that of a primitive using sub-graph isomorphism. The idea of the rule-based method for feature extraction is that rules are used to capture the knowledge about geometric and topological properties of form features. Features are recognized on the basis of certain pre-specified rules that are characteristic to the features. Volumetric approaches are based on the idea of finding the materials that must be removed from a raw stock to produce a part. Instead of relying on pattern matching like the techniques previously mentioned, this approach exploits convex hull algorithms and Boolean operations for feature analysis. The convex hull is the smallest convex set that contains the polyhedral object.

The classification or taxonomies of features is very importance to recognizing the features and depended on the context, the application that the model built to it. In this research we will deal with the polyhedral depression features such as slot, step, pocket, blind step, and blind slot and so on as primitive of simple features.

## 2. Literature review

All Automated Feature Recognition (AFR) algorithms include two important components: the definition of the features and the feature-recognition mechanism. In graph –based approach which is similar to proposed research, ***Joshi (1988)*** represents parts and definitions of features by an Attributed Adjacency Graph (AAG). An attribute value (zero or one) is assigned to each arc of the AAG depending on edge convexity or concavity. The graphs are stored as adjacency matrices internally. Joshi assumes that "a face that is adjacent to all its neighboring faces with

convex angles does not form part of a feature." Based on this assumption, the algorithm decomposes the AAG into sub- graphs by deleting the nodes which are only connected by convex edges. These sub graphs are further analyzed by the algorithm to determine the feature type [1].

***Ketan (1999)*** has developed the AAG. He suggests the Improved Attribute Adjacency Graph (IAAG) algorithm has the capability to the topological information in addition to the geometric relationships of faces that comprise the feature to overcome some shortcomings of AAG algorithm .He used values (+0,-0,0,1,+1,-1) in addition to the values (0,1) of the AAG [6]. In Volumetric approaches, ***Regli, Gupta and Nau(1994)*** they presents a methodology for recognizing a class of machinable features and addressing the computational problems posed by the existence of feature-based alternatives. He addresses a class of volumetric features that describe material removal volumes made by operations on 3-axis vertical machining centers including: drilling, pocket-milling, slot-milling, face milling, chamfering, filleting, and blended surfaces [7, 8,9].

In evidence-based reasoning approaches, ***Marefat (1997)*** combines and propagates evidences to determine a set of correct virtual links to be augmented to the cavity graph representing a depression of the object so that the resulting super graph can be partitioned to obtain the features of the object. The hierarchical belief network is constructed based on the hypotheses for the potential virtual links [10]. ***Ji and M.M. Marefat (2003)*** introduce an evidential reasoning-based approach for recognizing and extracting manufacturing features from solid model description of objects. They introduce a Dempster–Shafer approach for generating and combining geometric and topologic evidences to identify and extract interacting features. The main contributions of this research include introducing different classes of evidences based on the geometric and topologic relationships at different abstraction levels for effective evidential reasoning and developing the principle of association to overcome the mutual exclusiveness assumption of the Dempster–Shafer theory [3]**.**

***Dimov, Brousseau, and Setchi (2007)*** they presents a new hybrid method that facilitates the deployment of AFR systems in different application domains, the method includes two main processing stages: learning and feature recognition. During the learning stage, knowledge acquisition techniques are applied for generating feature-recognition rules and feature hints automatically from training data. Then, these hints and rule bases are utilized in the feature-recognition stage to analysis boundary representation (B-Rep) part models and identify their feature-based internal structure [11].

**3. Research Feature Taxonomy:-**

Rather than specifying all geometrical information that defines a feature type, it is possible to group features into hierarchical tree structure. These structures are commonly called feature taxonomies .The properties of the branch features in the structure are passed down to their leaf features [12].

Figure (1) shows the taxonomy of manufacturing features. In this research we deal with depression prismatic features. Use the Developed Attribute Adjacency Matrix DAAM algorithm to recognize the features type. We have developing Rules as knowledgebase to parsing the feature types for prismatic part.

### 3.1 Attribute Adjacency Graph (AAG) and Edges Definition:-

The attribute adjacency graph (AAG) can be defined as a graph G = (N,A,T) where N is the set of nodes, A is the set of arcs, and ,T is the set of attributes to arcs in A, such that:

- For every face f in F, there exists a unique node n in N.
- For every edge e in E, there exists a unique arc in A, connecting the nodes $n_i$ and $n_j$, corresponding to face $f_i$ and face $f_j$, which share the common edge e.

For every arc a in A, is assigned an attribute t, t= o, if the faces share the edge from a concave angle and t=1, if the faces share the edge from a convex angle.

Let *e* be the edge shared between two faces Fi and Fj figure (2). Then, convexity or concavity of the edge is found as follows[13].

$Q=\cos^{-1}((fi.\ f_j)/|fi.\ f_j|)$

Where:

fi and fj are normals of the faces Fi and Fj

Q is the angle between Fi and Fj

If:

Q=90, then the relation is concavity (0).

Q<90, then the relation is concavity (-0).

Q>90, then the relation is concavity (+0).

Otherwise the relation is convexity (1) or no relation between Fi and Fj the relation is empty (*).

### 3.2 Constructing Developed Attribute Adjacency Matrix (DAAM) and Extraction and Recognition for Isolated Features:

The steps of algorithm for construct the DAAM and feature recognition :

Step- 1: Construct the whole part matrix (N, M) which is equal to the number of faces in part, and then find the adjacencies faces to arbitrary face which is selected .If in analysis an adjacency is found between two faces then it is documented by entering (0, +0,-0, 1) into the cell (Ni, Mj) of the matrix which represents the row and column number. The procedure is continued until the whole part is analyzed; the flowchart of this analysis is illustrated in Figure (3).

Step- 2: By examination of the column for any face in DAAM, if all cells in column contain (1) or empty (*); then this face dose not represent cluster, otherwise if the column contains (0),(-0) or (+0) in any cell , then this face represents cluster and the cluster corresponding to the face which has relation to analysis face is added.

$$w_j=\{u \cup w_i\} \qquad \text{…….(1)}$$

And this continues until the all faces are examined. The results from this step represent the set of clusters. The flowchart of this step is shown in Figure (4).

$$w_j=\{w_j,w_{i+1},w_{i+2},\ldots,w_n\} \qquad \text{……...(2)}$$

Step- 3: From the results in previous step by examine any cluster do the following:

a) W=Φ ( empty set) ,then end

,if no, do the following :

b) $w_{o+1}= w_o$ U the underlying set of cluster $w_o$ corresponding to the clusters from step 2 …….(3)

c) If $w_{o+1} – w_o= \Phi$, then $F_r = w_o$ and delete the underlying set for any set in $w_o$ ( W – every underlying set in $w_o$ ) from (a) because any face lies only in one cluster due to applying isolated features . If no, repeat (b).The results from this step are set of clusters:

$F_r = w_o$, $F_{r+1} = w_o$, $F_{r+2} = w_o$ …..(4)

These clusters represent explicitly set of features and these results are used in next step. The flowchart of this step is shown in Figure (5).

Step- 4: For each cluster $F_r$ examine this cluster to recognize feature type corresponding to it, by doing the following:

a) By taking every column corresponding to the underlying set in Fr. Calculate the number of the cells contained p= (0), S= (-0) and A= (+0) and then match to the feature pattern to know the type of this feature.

b) After calculating the p, S, A for all columns corresponding to the underlying set in Fr (the columns in DAAM).

c) Calculate H (the number of the cell contains 0,-0, +0) (H= P/2+S/2+A/2).

Figure (6) shows a flowchart of feature recognition for some feature types.

A simple example is illustrated in Figure (7) to examine the feature extraction and recognition algorithm for simple slot feature.

Step- 1: First build DAAM by identifying the edge situations for every face in part model as shown in Figure (8), step-1.

Step- 2: In step 2 of the algorithm, the results of this step are:

$w_1=\{w_1,w_3,w_5\}$,$w_3=\{w_3,w_1\}$,$w_5=\{w_5,w_1\}$,as shown in figure (8), step-2

Step- 3: By taking the results in pervious step and then examining the results in step3 of the algorithm. Take the cluster $w_3=\{w_3,w_1\}$ because the algorithm does not accept the chance for taking any cluster.

$w_o=w_3 U w_1$.

$w_{o+1}=\{w_1,w_3,w_5\}$.

If $w_{o+1}- w_o =\Phi$ ; then $F_1= w_o=\{w_1,w_3,w_5\}$, and delete the cluster $w_1,w_2,w_5$ in step 2 and repeat (b) in step 3. The result of this example is : $F_1=\{w_1,w_3,w_5\}$. As shown Figure (8), step-3

Step- 4: This cluster is applied to step 4 in the algorithm by taking the columns corresponding to $w_1$,$w_3$ and $w_5$, in this example , the number of the cells contains (0) is four and the number of the cells which contain (+0) or (-0) is zero. The type of feature in this example is simple slot.as shown in Figure (8), step-4.

**4. Experiment results**

The feature recognition interfaced with AutoCAD 2002 package. AutoCAD stores the representation of the solid in form of the B-Rep, pre-process need to convert the DXF file format to obtain the representation of the part in form of faces, edges and vertices. The algorithm coded in visual basic 6 programming language and implemented on Pentium IV. Several parts with varying number of

features were tested. Figure (9) show the part with isolated features.

Applied the proposed algorithm we recognize six primitive clusters witch are representing features implicitly as shown in figure (10).

**4.1. Another example:**

Applied the algorithm on another example shown in figure (11), we recognize seven primitive clusters witch are representing features implicitly as shown in figure (12).

**5. Conclusion**

The main contributions of this research include building the adjacent relations between faces of the part by geometrical and topologies definitions in form of matrix, and use clustering operation to extract the exact faces of feature which simplifies the procedure of search and use different classes of geometrical characteristics. The system was success to recognize several primitive features for prismatic part with high efficiency. The feature recognition is useful in variety of manufacturing application such as computer process planning and dimensional inspection planning where the feature information can be used to determine the machining sequence and tool selection and tool approach direction. Extracting and recognizing different types of protrusion and cylindrical features and several types of interacting features are directions for future works.

**References**

[1] S. Joshi and T.C. Hang "Graph-based heuristic for Recognition of Machined Features from a 3D solid Model" Computer-Aided Design, Vol. 20, No. 2, March 1988.

[2] M.R. Henderson, Computer recognition and extraction of form features: a cad/cam link, Comput. Ind. 5 (1984) 329–339.

[3] Q. Ji and M.M. Marefat "A Dempster–Shafer approach for recognizing machine features from CAD models" Pattern Recognition, Vol.36, PP.1355-1386. 2003.

[4] Q. JI and M. M. Marefat "Machine Interpretation of CAD Data for Manufacturing Applications", ACM Computing Surveys, Vol. 24, No. 3, September 1997.

[5] A. Kusiak "Intelligent Manufacturing Systems" Book, University of Lowa, 1990.

[6] H. S. Ketan "Integrating Design and Inspection Activities Using AI Technique" Ph.D, Thesis, U.O.T, 1999.

[7] W. C. Regli, S. K. Gupta and D. S. Nau "Extracting Alternative Machining Features: An Algorithm Approach" Technical Report ISR-TR 94-55, July, 1994.

[8] W. C. Regli, S. K. Gupta and D. S. Nau "Manufacturing Feature Instances: Which one to Recognize?" University of Maryland, Technical Report, 1994.

[9] W. C. Regli, S. K. Gupta and D. S. Nau "An Application of Distributed Solid Modeling: Feature Recognition" University of Maryland, Technical Report ISR, 1994

[10] M. M. Marefat and Q. Ji "Hierarchical Bayesian Methods for Recognition and Extraction of 3-D Shape Features from CAD Solid Models" IEEE Transactions on Systems and Humans, Vol. 27, No. 6, November 1997.

[11]S. S Dimov, E. B Brousseau, and R. Setchi "A hybrid Method for Feature Recognition in Computer-Aided Design Model" Proc. IMechE

Vol. 221, Part B: J. Engineering Manufacture, 2007

[12] S. B. Barwick and A. Bowyer "Feature Technology" University of Bath, Technical Report, January 1993. -TR 94-82, 1994..

[13] M. P. Bhandarkar, R. Nagi "STEP-based feature extraction from STEP geometry for Agile Manufacturing" Computers in Industry Vol. 41, PP 3–24, 2000

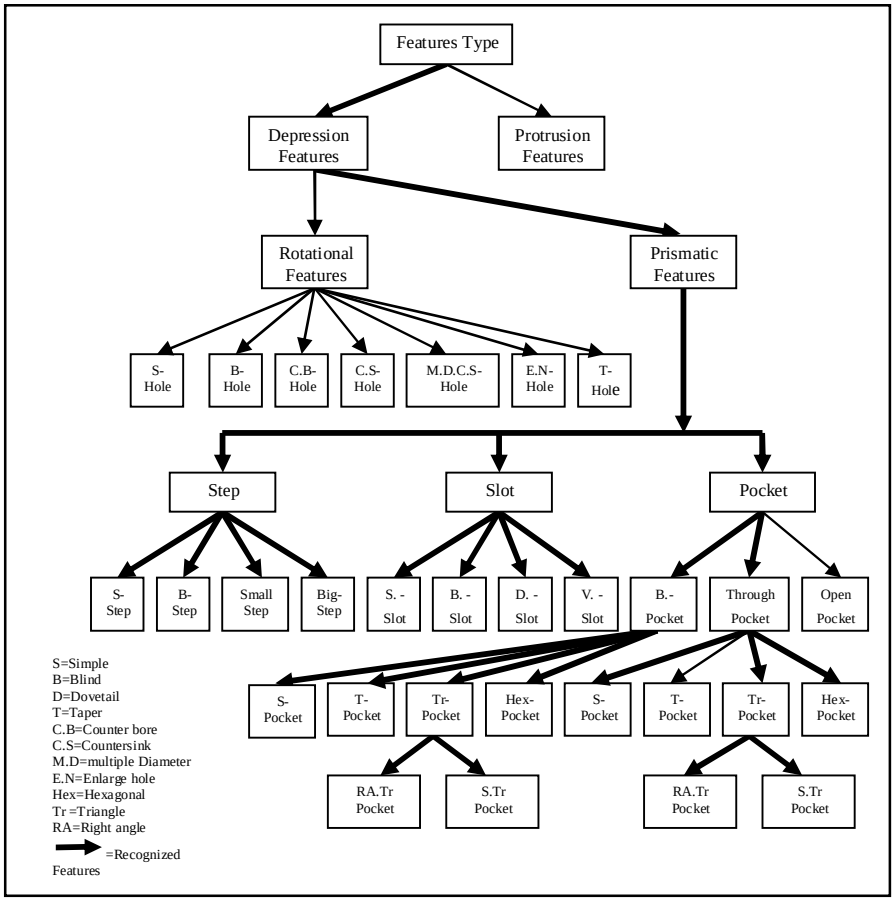



**Figure (1) Manufacturing Features Taxonomies [6].**

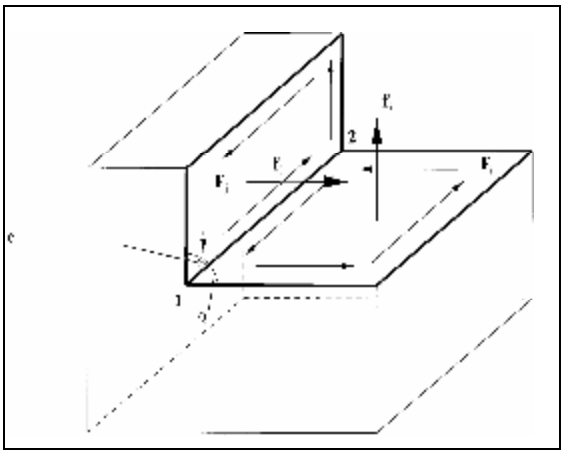


**Figure (2) Establishing the convexity or concavity of the edge**

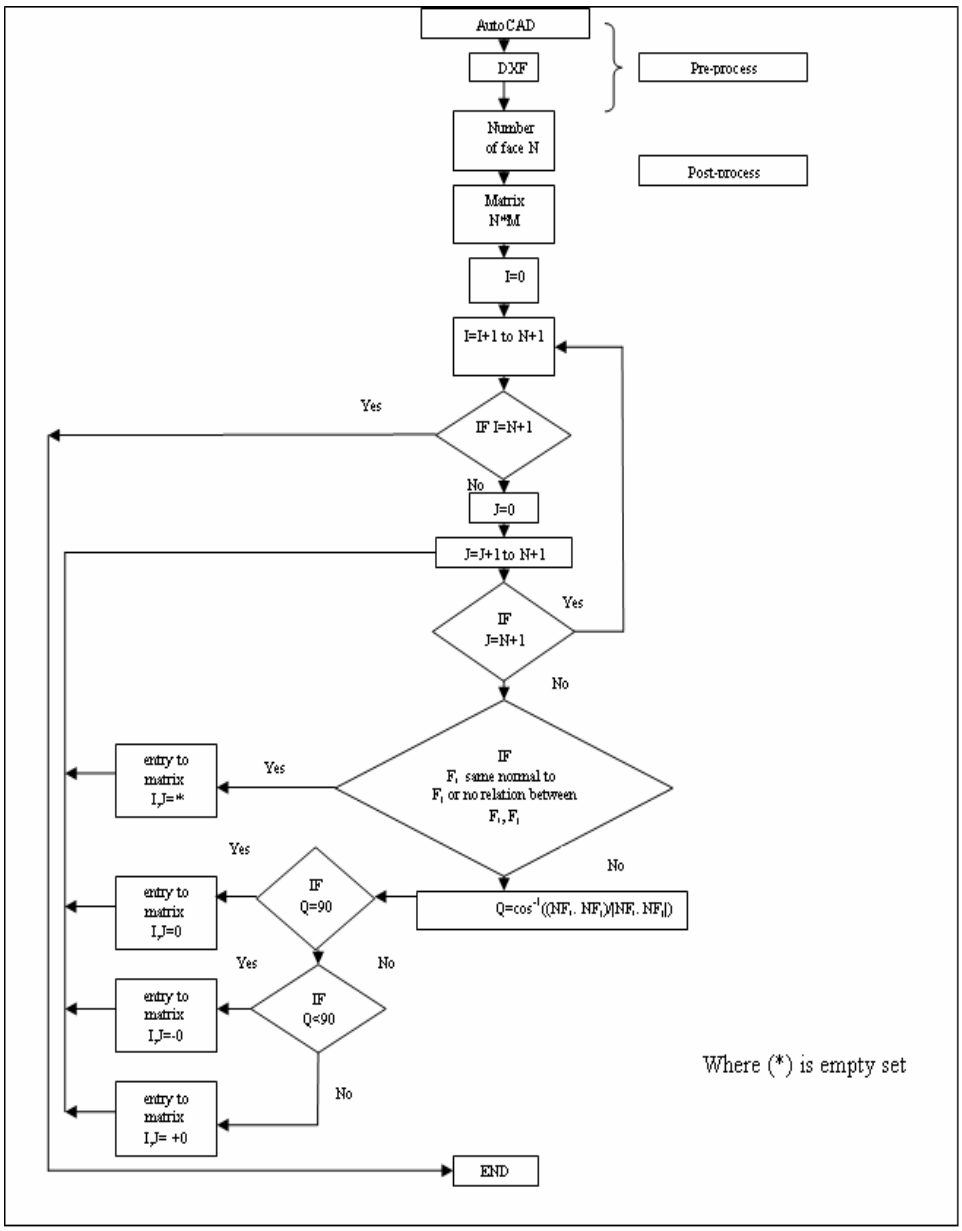



**Figure (3) Flowchart for constructing Developed Attribute Adjacency Matrix**

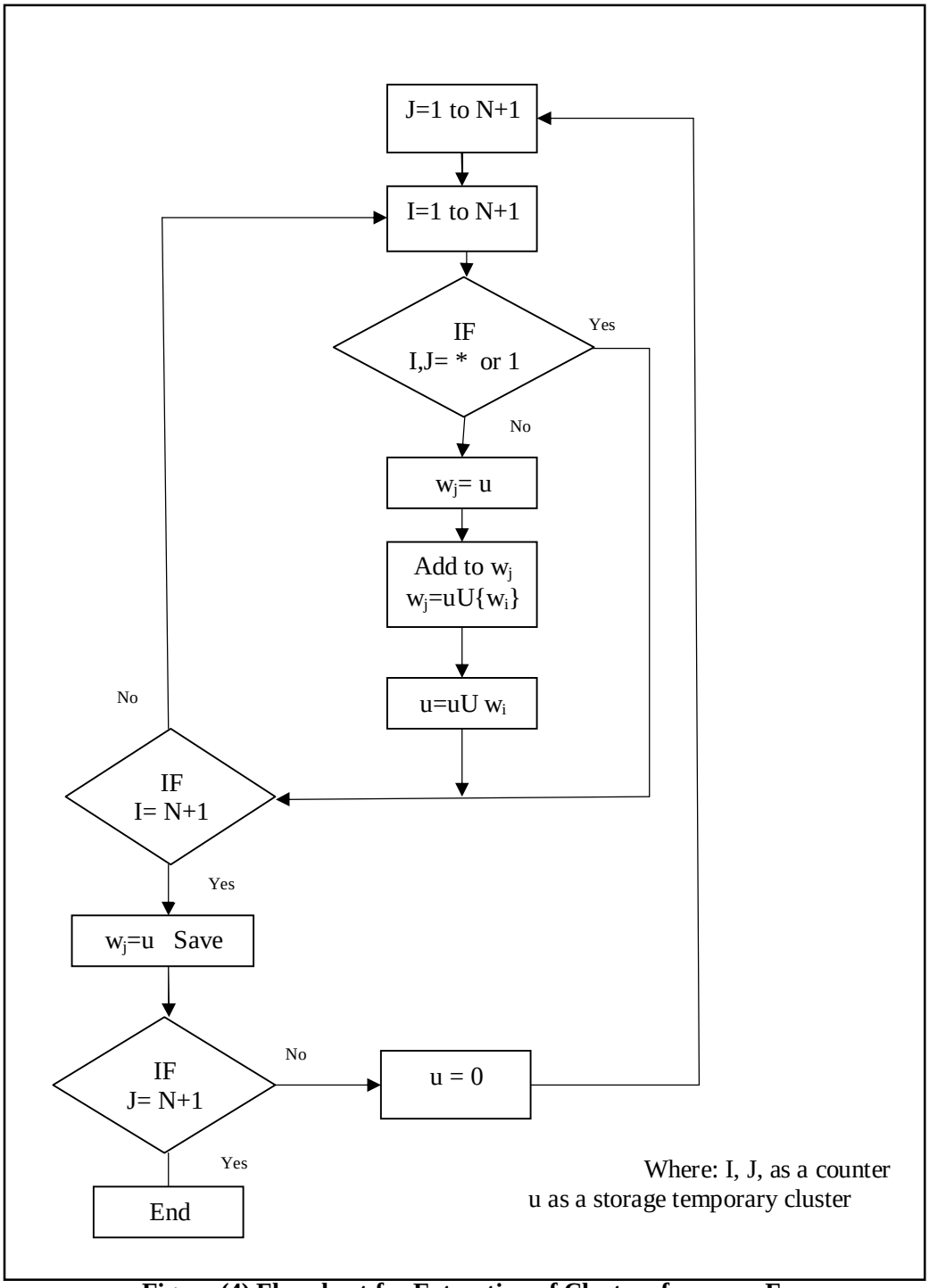



**Figure (4) Flowchart for Extraction of Clusters for every Face.**

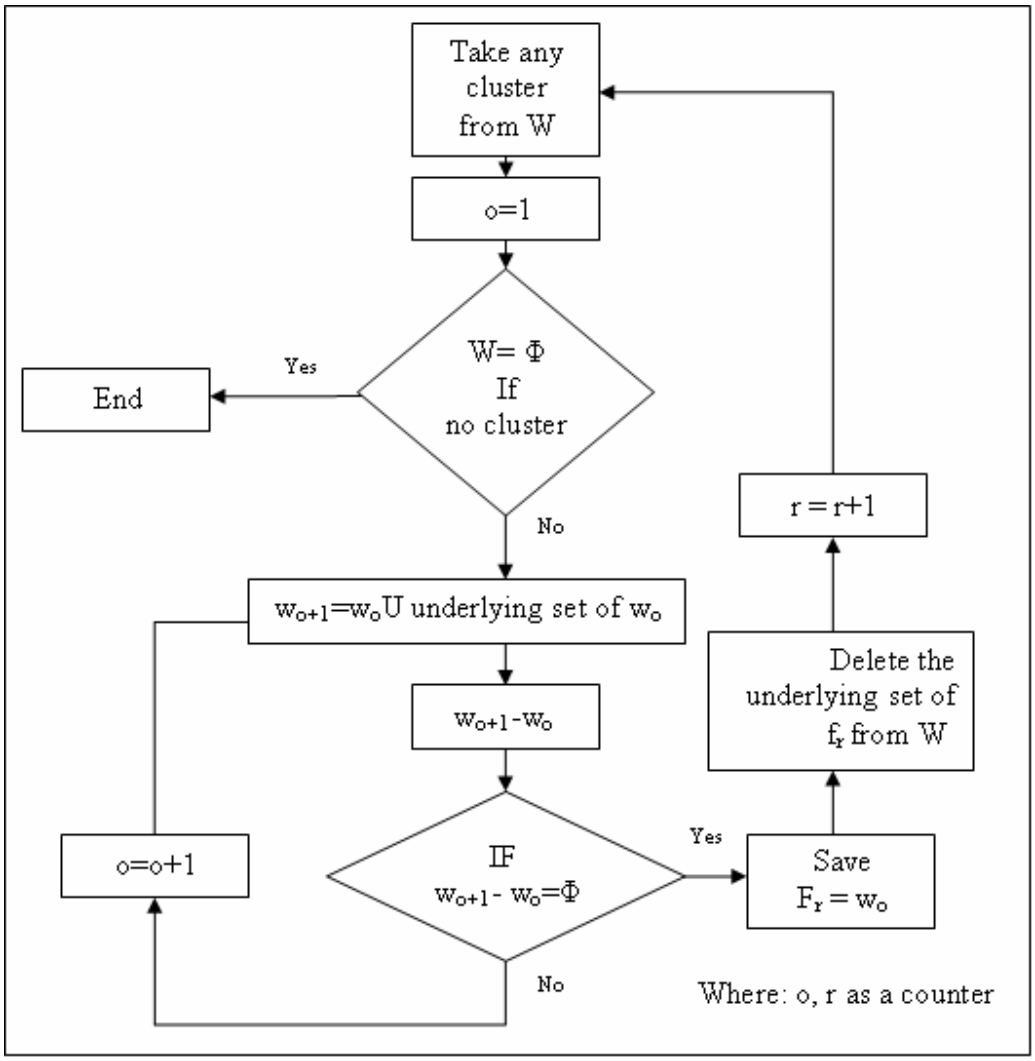



**Figure (5) Flowchart for explicit set of Features.**

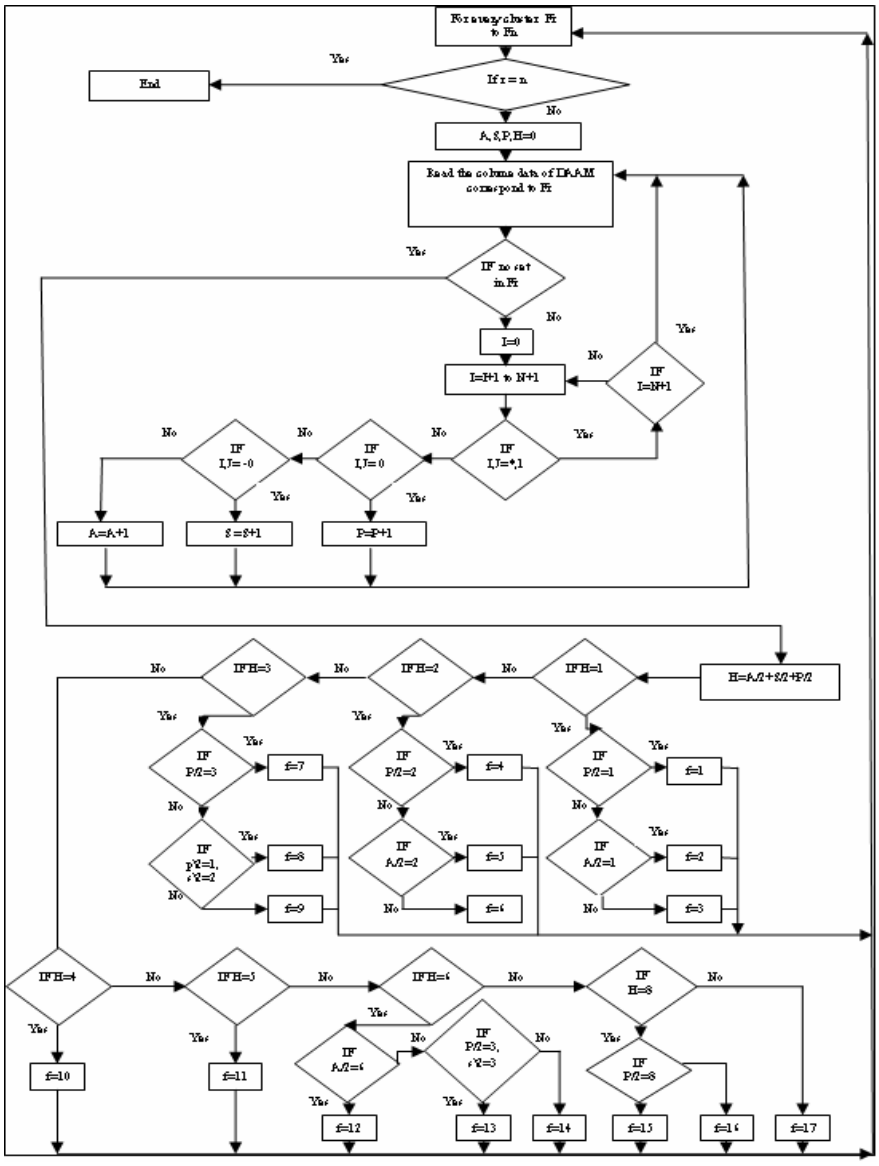



**Figure (6) Flowchart for Recognition some Types of Features.**

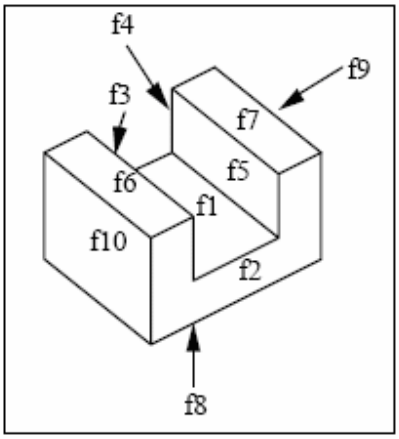



**Figure (7) A simple slot feature**

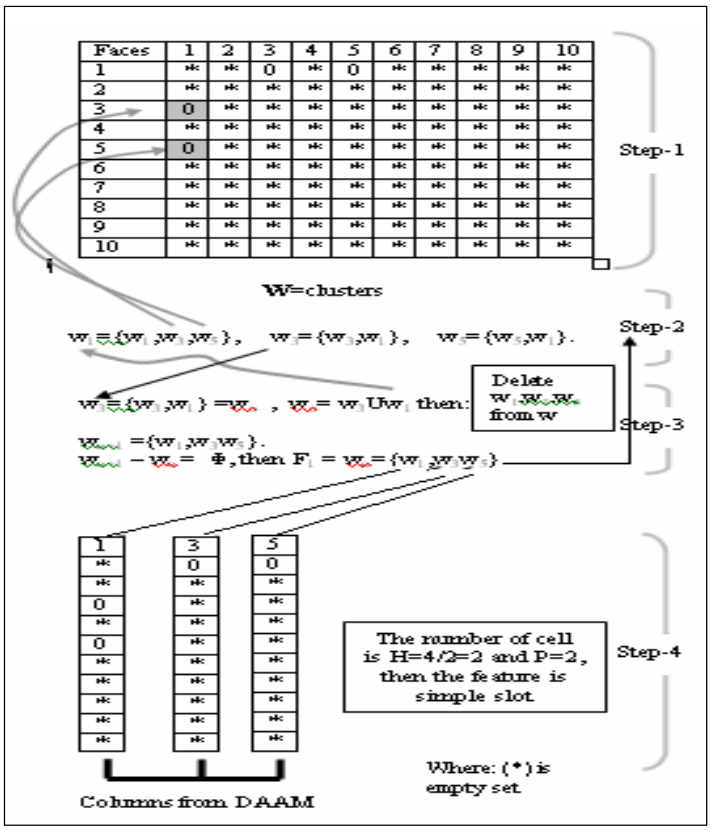



**Figure (8) Feature recognition algorithm for a simple slot.**

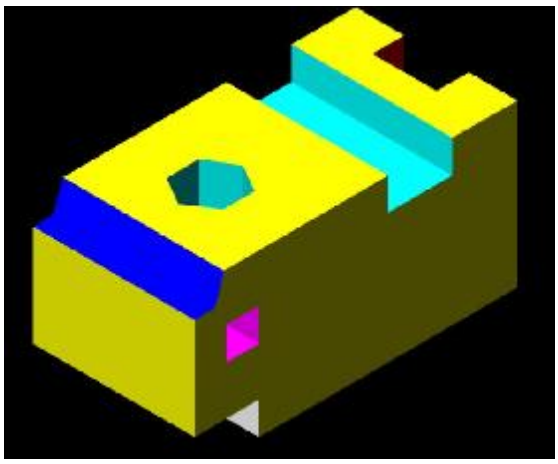
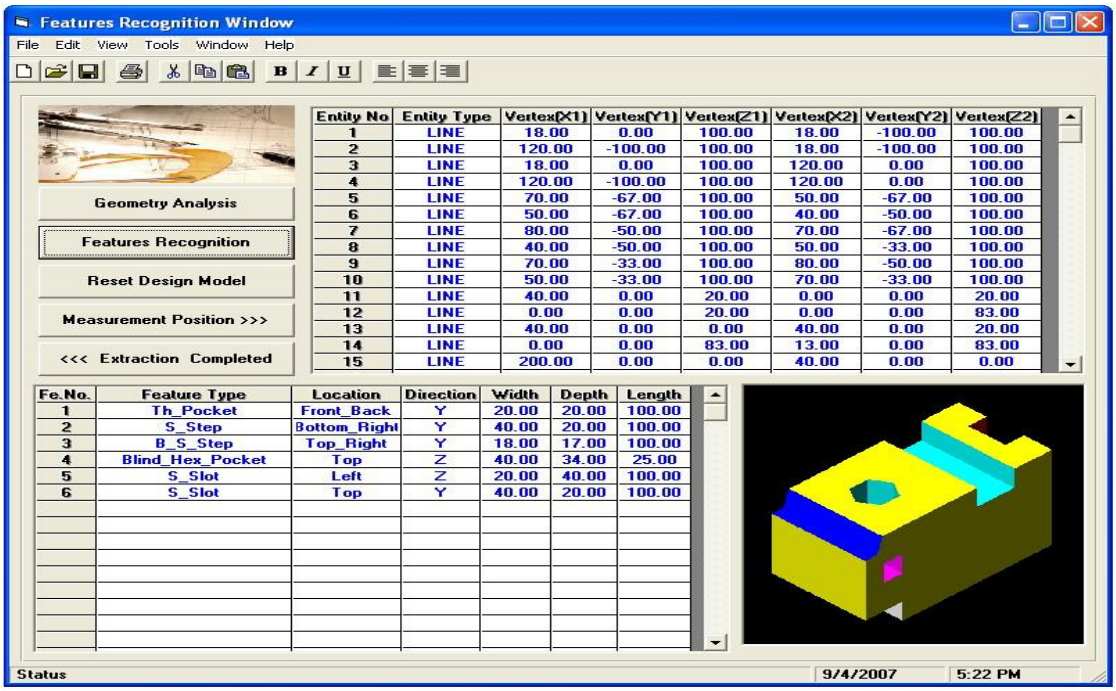


**Figure (9) Test part with isolated features.**




| Entity No | Entity Type | Vertex(X1) | Vertex(Y1) | Vertex(Z1) | Vertex(X2) | Vertex(Y2) | Vertex(Z2) |
|---|---|---|---|---|---|---|---|
| 1 | LINE | 18.00 | 0.00 | 100.00 | 18.00 | -100.00 | 100.00 |
| 2 | LINE | 120.00 | -100.00 | 100.00 | 18.00 | -100.00 | 100.00 |
| 3 | LINE | 18.00 | 0.00 | 100.00 | 120.00 | 0.00 | 100.00 |
| 4 | LINE | 120.00 | -100.00 | 100.00 | 120.00 | 0.00 | 100.00 |
| 5 | LINE | 70.00 | -67.00 | 100.00 | 50.00 | -67.00 | 100.00 |
| 6 | LINE | 50.00 | -67.00 | 100.00 | 40.00 | -50.00 | 100.00 |
| 7 | LINE | 80.00 | -50.00 | 100.00 | 70.00 | -67.00 | 100.00 |
| 8 | LINE | 40.00 | -50.00 | 100.00 | 50.00 | -33.00 | 100.00 |
| 9 | LINE | 70.00 | -33.00 | 100.00 | 80.00 | -50.00 | 100.00 |
| 10 | LINE | 50.00 | -33.00 | 100.00 | 70.00 | -33.00 | 100.00 |
| 11 | LINE | 40.00 | 0.00 | 20.00 | 0.00 | 0.00 | 20.00 |
| 12 | LINE | 0.00 | 0.00 | 20.00 | 0.00 | 0.00 | 83.00 |
| 13 | LINE | 40.00 | 0.00 | 0.00 | 40.00 | 0.00 | 20.00 |
| 14 | LINE | 0.00 | 0.00 | 83.00 | 13.00 | 0.00 | 83.00 |
| 15 | LINE | 200.00 | 0.00 | 0.00 | 40.00 | 0.00 | 0.00 |

| Fe.No. | Feature Type | Location | Direction | Width | Depth | Length |
|---|---|---|---|---|---|---|
| 1 | Th_Pocket | Front_Back | Y | 20.00 | 20.00 | 100.00 |
| 2 | S_Step | Bottom_Right | Y | 40.00 | 20.00 | 100.00 |
| 3 | B_S_Step | Top_Right | Y | 18.00 | 17.00 | 100.00 |
| 4 | Blind_Hex_Pocket | Top | Z | 40.00 | 34.00 | 25.00 |
| 5 | S_Slot | Left | Z | 20.00 | 40.00 | 100.00 |
| 6 | S_Slot | Top | Y | 40.00 | 20.00 | 100.00 |

**Figure (10) The output results from feature recognition algorithm**

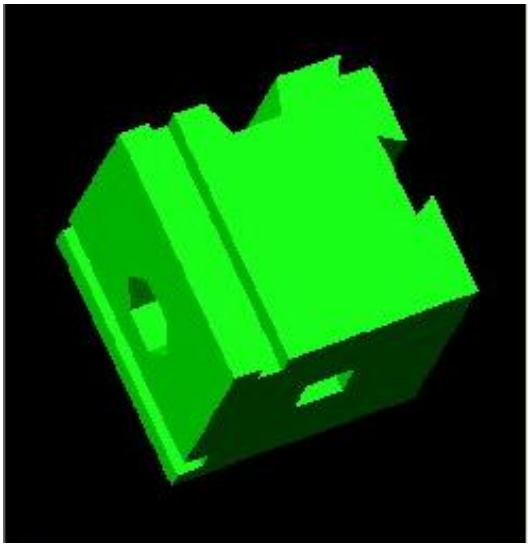
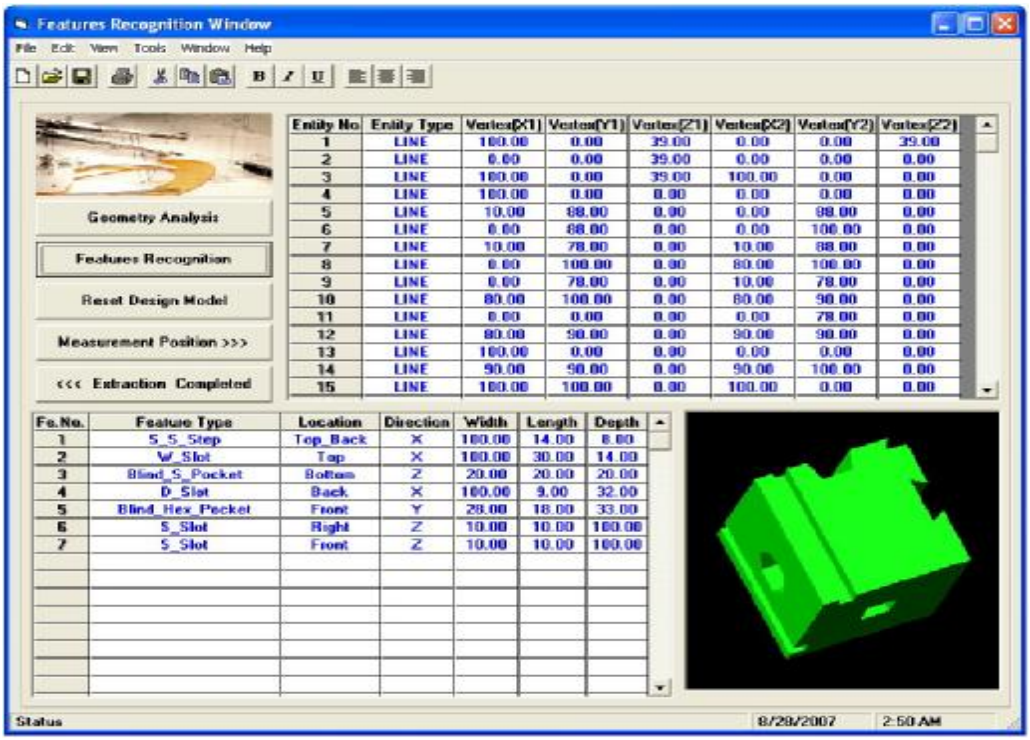


**Figure (11) Another part with isolated features.**




| Entity No | Entity Type | Vertex(X1) | Vertex(Y1) | Vertex(Z1) | Vertex(X2) | Vertex(Y2) | Vertex(Z2) |
|---|---|---|---|---|---|---|---|
| 1 | LINE | 100.00 | 0.00 | 39.00 | 0.00 | 0.00 | 39.00 |
| 2 | LINE | 0.00 | 0.00 | 39.00 | 0.00 | 0.00 | 0.00 |
| 3 | LINE | 100.00 | 0.00 | 39.00 | 100.00 | 0.00 | 0.00 |
| 4 | LINE | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | LINE | 10.00 | 88.00 | 0.00 | 0.00 | 88.00 | 0.00 |
| 6 | LINE | 0.00 | 88.00 | 0.00 | 0.00 | 100.00 | 0.00 |
| 7 | LINE | 10.00 | 78.00 | 0.00 | 10.00 | 88.00 | 0.00 |
| 8 | LINE | 0.00 | 100.00 | 0.00 | 80.00 | 100.00 | 0.00 |
| 9 | LINE | 0.00 | 78.00 | 0.00 | 10.00 | 78.00 | 0.00 |
| 10 | LINE | 80.00 | 100.00 | 0.00 | 80.00 | 90.00 | 0.00 |
| 11 | LINE | 0.00 | 0.00 | 0.00 | 0.00 | 78.00 | 0.00 |
| 12 | LINE | 80.00 | 90.00 | 0.00 | 90.00 | 90.00 | 0.00 |
| 13 | LINE | 100.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 14 | LINE | 90.00 | 90.00 | 0.00 | 90.00 | 100.00 | 0.00 |
| 15 | LINE | 100.00 | 100.00 | 0.00 | 100.00 | 0.00 | 0.00 |

| Fe.No. | Feature Type | Location | Direction | Width | Length | Depth |
|---|---|---|---|---|---|---|
| 1 | S_S_Step | Top_Back | X | 100.00 | 14.00 | 8.00 |
| 2 | W_Slot | Top | X | 100.00 | 30.00 | 14.00 |
| 3 | Blind_S_Pocket | Bottom | Z | 20.00 | 20.00 | 20.00 |
| 4 | D_Slot | Back | X | 100.00 | 9.00 | 32.00 |
| 5 | Blind_Hex_Pocket | Front | Y | 28.00 | 18.00 | 33.00 |
| 6 | S_Slot | Right | Z | 10.00 | 10.00 | 100.00 |
| 7 | S_Slot | Front | Z | 10.00 | 10.00 | 100.00 |

**Figure (12)The output results from feature recognition algorithm.**