

**Military Technical College  
Kobry El-Kobbah,  
Cairo, Egypt.**



**16<sup>th</sup> International Conference  
on Applied Mechanics and  
Mechanical Engineering.**

## **FEATURE RECOGNITION TECHNIQUE FOR MILLING OPERATION**

S.E. Sarhan<sup>\*</sup>, A. A. Afifi<sup>\*\*</sup> and M.S. Abbas<sup>\*\*\*</sup>

### **ABSTRACT**

This paper presents a methodology for implementing the feature recognition system for achieving the Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) integration goals. Feature-recognition technology is used for automating the extraction of data from CAD product models to minimize redundant user interaction with a product model. The feature-recognition process was developed using data extracted from DXF file (points, lines, arcs,...) or rule-based methods with wire-frame geometry extracted from an IGES neutral file format ( lines, axis, arcs,...). Use of wire-frame models simplifies product geometry and has the potential to support rapid manufacturing shape evaluation at the conceptual design stage.

The developed feature recognition technique involves the reconstruction of the part design features, in the form of manufacturing features (11 manufacturing features). Output data are displayed to the user as design features, and manufacturing features with all required data technique for manufacturing. A case study is given to demonstrate, and verify the proposed algorithm.

### **KEY WORDS**

Feature Recognition, Computer aided part programming, CAPP, CIM.

---

\* Lecturer assistant, Dept. of Design and Production, German University, Cairo, Egypt.

\*\* Associate professor, Dept. of Design and Production, German University, Cairo, Egypt.

\*\*\* Professor, Dept. of Mechanical Engineering, Banha University, Cairo, Egypt

## INTRODUCTION

Feature recognition is the ability to automatically or interactively identify and group topological entities into functionally significant features such as holes, slots, pockets, fillets, ribs etc. In order to integrate CAD and CAM an interface for neutral (mostly geometric) data exchange is needed between CAD and CAM. Many data exchange formats have been developed in the past, their primary purpose is to exchange geometric data. Most widely used data exchange formats like IGES, DXF and PDES will transfer only the geometric information and they may not support the data throughout the life cycle [1].

In recent years, various researchers have come up with different ways and means to integrate CAD and CAM. Automatic feature recognition from CAD solid systems highly impacts the level of integration [2].

Milling parts have usually a large number of features. Feature recognition is not only to recognize the individual features of the workpiece, but also to recognize the different combinations among them to be an efficient tool in computer aided operation planning system and consequently in computer aided part programming techniques.

According to the survey of the different feature recognition systems (IFRM system, IF2, ANN, systems developed by Mangesh P. Bhandarkar, Rakesh Nagi, T. Derili, H. Filiz, Y. Woo', E. Wang', Y. S. Kim', H. M. Rho<sup>2</sup>, 1Chee Fai Tan, 1V.K. Kher and 2N. Ismail, T. J. Jones<sup>1</sup>, C. Reidsema<sup>1</sup> and A. Smith<sup>2</sup>; D. Sreeramulu, C.S.P. Rao) [2,3,4,5,6,7,8,9,1] and computer aided process planning systems (, CADEXCATS, OPPS-ROT, MASCAPP, IPIM, PLM/CAPP, CADLOG, ICAPP, FBAPP, OSCAP, IDEF, GISCAPP) [10,11,12,13,14,15,16,17,18,19,20] it has been found out that there still exist certain points, which have not been covered sufficiently. Flexibility of the most systems is considerably low due to few numbers of predefined features and most systems deals with single type of CAD systems either 2D or 3D system.

In the present work, 2D and 3D CAD models are used for the determination of the necessary machining features for prismatic parts which manufactured by milling machine.

## SYSTEM REQUIREMENTS

The software program is developed to work under Microsoft windows XP operating system or later. The main software program consists of several modules, which are developed under different programming, drafting and modelling (AutoCAD package to be used for drafting of input part drawing and PROENGINEER for solid modelling). Visual basic 6 package has been used for constructing the system engine, and user interfaces (input and output).

## SYSTEM CONSTRAINTS

There is no constraints in introducing the workpiece 3D solid modelling but user has

to present the workpiece in two views ( front and top views) for using 2D system.

## DATA EXTRACTION MODULE

A technique of data extraction has been developed during the course of this research to read the DXF file or IGES file, this technique is summarized in the following steps:

A loop was developed to read all the information from the CAD file required to build the machining features.

The loop has been developed to read only some entities which will be used in the feature recognition module. These entities are Lines, arcs, circles, dash lines and text in case of using DXF file or lines, arcs, axis in case of using IGES file.

The DXF/IGES file reading module reads those entities using a counter which is moving all over the entities section of the DXF file and extracts the required data to build the manufacturing features.

### Extraction of the Start and End Point of an Arc

One of the problems facing the software data extraction module is the extraction of the start and end point of an arc. A methodology has been developed to find these points. The arc is defined in the DXF file by the radius of arc, the values of center point in both X and Y axes, the angle of the start point, and the angle of the end point. A technique has been developed in order to calculate the values of X and Y axes for both start and end points. These values will be used later in part programming module.

The following equations are used to get the arc's start and end points.

$$X_s = X_c + R * \cos\theta_s.$$

$$X_e = X_c + R * \cos\theta_e.$$

$$Y_s = Y_c + R * \sin\theta_s.$$

$$Y_e = Y_c + R * \sin\theta_e.$$

where:

$X_c$ : Distance between the center of arc circle and the origin on X-Axis.

$Y_c$ : Distance between the center of arc circle and the origin on Y-Axis.

$R$ : radius of circle of arc.

$X_s$ : Distance between the start point of arc and the origin on X-Axis.

$X_e$ : Distance between the end point of arc and the origin on X-Axis.

$Y_s$ : Distance between the start point of arc and the origin on Y-Axis.

$Y_e$ : Distance between the end point of arc and the origin on Y-Axis.

$\theta_s$ : Start angle of arc.

$\theta_e$ : End angle of arc.

All these parameters are shown in figure (3).

### Elevation and Plan Views Separation Technique from DXF File

The data for both views (elevation and plan) are in X and Y axes values but the data for actual part should be in X, Y and Z values so we have to separate the two views to be one in X-Y values and the other to be in X-Z value.

The separation between elevation view and plan view data to enable the software feature recognition module of building the machining features. In order To separate the two plans a search has been made to find the (separation point) in the elevation view and this point is shown in figure (1).

In order to find the separation point a methodology has been developed according to the following steps. The loop searches for the entity having maximum value of Y axis then it continues to find entity having the value of Y less than the first one and at the same time is linked to it, the search is continued in the same principle as shown in figure (2).

The loop continues in searching and linking till it finds an entity that is not connected with a line or an arc having a value in y less than it, then it records the value of Y this entity as the separation point value.

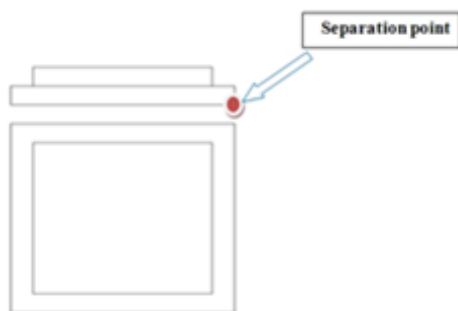


Figure (1)

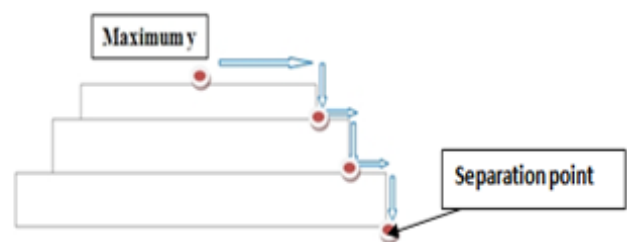


Figure (2) the separation point foundation loop

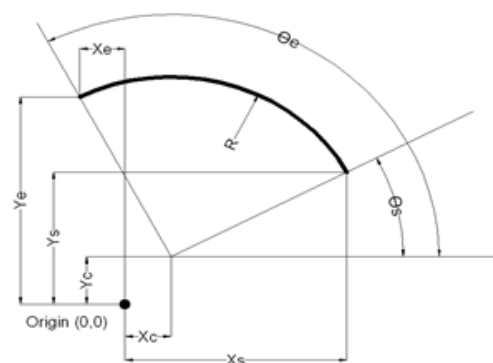


Figure (3) Arc parameters

## Data Transformation from (X-Y) to (X-Z) Module

All the entities having Y value equal greater than Y value of the separation point are recorded in the database in the tables of elevation view (X-Z), and simultaneously all the entities having Y value less than the separation point are recorded in the database in the tables of plan view (X-Y).

## Determination of Sketched Plane from IGES File

one of the complicated tasks in the reading of the three dimensional prismatic parts is the determination of the sketched plane as the user can sketch his part on front, right or top plane.

In order to determine the sketched plane we find that if the value of Z direction is equal to zero that means the user sketch on the front plane and if the value of Y direction is equal to zero then the user sketch on the top plane else the user sketch on the right plane.

The developed software can transform any sketched plane to be front plane which means that all depth will be in Z axis while width and length will be in X and Y axes.

## Shifting Data to Origin

As a facility to the drafting and modeling process a methodology was developed to enable the user to start his/her drawing from any point in the screen (2D system) or any point and any plane in the screen (3D modeling system). Then the developed software will read the data and offset it to origin point to ease the computer aided part programming methodology. Data shifting methodology depends on developing a loop to search for the value of  $X_{min}$  and  $Y_{min}$  in all lines and arcs data (2D system) or  $X_{min}$ ,  $Y_{min}$  and  $Z_{max}$  (3D modeling system), and then subtract its values from all the points. Then the result from this process is the data calculated from the origin point (0, 0) in 2D drawing or (0, 0, 0) in 3D drawing.

## FEATURE RECOGNITION MODULE

The feature recognition module depends mainly on the data extracted from data extraction module. The feature recognition module processes the extracted data and builds the machining features.

The developed software using smart loop that produces the entire feature with all the data required for process plan or part program generation. Merging all these incoming modules is a very complex process, but it helps in increasing the accuracy of the output.

## Recognition of Raw Material Dimensions

- **Using DXF file or IGES file**

The developed software can calculate the work piece dimension by calculating the

difference between maximum and minimum X, Y and Z which give us the length, width and thickness of workpiece.

### Recognition of Normal External Contour

- **Using DXF file**

The recognition module of any external contour using DXF file has been developed in the following steps:

1. Loop has been generated to search in the database for a line or an arc that starts with (0, 0) point which should exist on a line or an arc after the data shift procedure.
2. After (0, 0) point is found the loop connects it with the next line or arc by its end point.
3. The loop keeps on searching for the ends and link them with the starts of the next entities (lines or arc) until it reaches the starting point again.
4. After the contour is recognized the depth of the contour is found by searching in contour data for the point ( $X_{min}$ ) and ( $X_{max}$ ) and then search in lines table in (X-Z) plane for the point corresponding to both as shown in figure (4).
5. After all the external contour data is recorded in external table in database all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.
6. After the recognition of the external contour, all the incoming contours (unless it was inner contour) are recorded in the data base of the external contour.
7. The software is able to recognize as many external contours as possible, and then record them in the external contour database giving them a respective order.

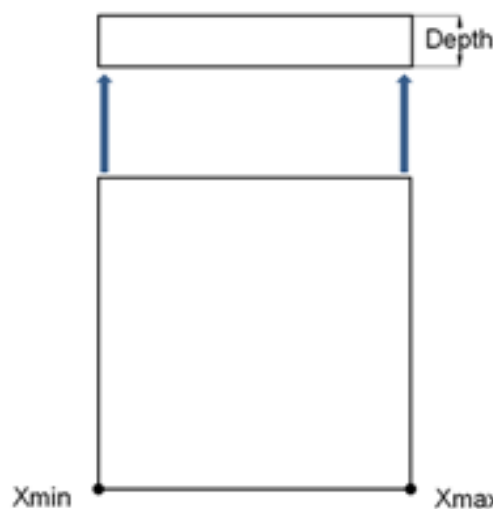


Figure (4) searching for contour data using DXF file

- **Using IGES file**

1. Firstly we need to divide all lines into three groups ( Horizontal lines , Vertical lines and Inclined lines) by using a loop which search in all lines to find the lines which have start and end points with same value in X and Y axes and different value of Z axis and record them in vertical line table. Then the loop will searches for the lines which have start and end points with same value in Z axis and have different values of X and Y axes and records them in horizontal line table. Any line has not any of the two previous conditions will be recorded in the inclined line table as shown in figure (5).
2. Loop will be created to search in the database for minimum value of Z axis in all horizontal lines and all arcs.
3. The loop searches in the database for a line or an arc that starts with  $(0, 0, Z_{min})$  point which should exist on a line or an arc after the data shift procedure.
4. After  $(0, 0, Z_{min})$  point is found, the loop connects it with the next line or arc by its end point.
5. The loop keeps on searching for the ends and link them with the starts of the next entities (lines or arc) until it reaches the starting point again.
6. After the contour is recognized the depth of the contour is found by searching in contour data for the point  $(X_{start}, Y_{start}, Z_{min})$ , cf. figure (6), and then search in vertical line table for a line has the same point as a start or an end point.
7. After the recognition of the external contour, all the incoming contours (unless it was inner contour) are recorded in the data base of the external contour.
8. After all the external contour data is recorded in external table in database all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.
9. The software is able to recognize as many external contours as possible, and then record them in the external contour database giving them a respective order.

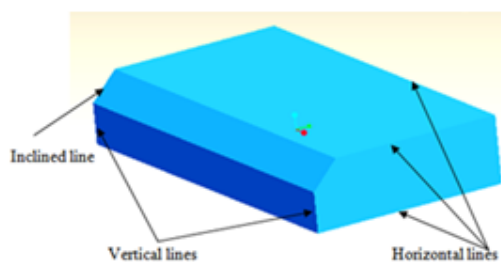


Figure (5) Determination of horizontal, vertical  
And inclined lines using IGES file

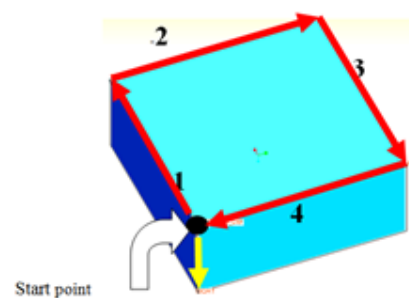


Figure (6) Searching for contour data  
Using IGES file

**The output data from the recognition of this feature which can be used later are:**

- The start and end point of each element of any external contour.
- Recognize the Type of elements (line or arc).
- Contour depth.
- Multiple external contours can be recognized and saved in database.

## Recognition of External Circular Contours

### • Using DXF file

1. Recognition loop searches in (X-Y) plane for all the circles and then record the diameter and  $(X_{center}, Y_{center})$  points for the first circle.
2. The loop then turns into searching for the projection line of this circle in (X-Z) plane table having a point with X value equal to  $(X_{center} + \text{CIRCLE radius})$  and assure that this projection line is a continuous line [ not dash line]
3. The depth of the circular contour is determined from the length of the projection line.
4. Then the loop records the depths in the circular contour table.
5. The steps from 1 to 4 are repeated until the software records all external circular contours.
6. After all the external circular contours data is recorded in external contours table in database, all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.
7. The software is able to recognize as many external circular contours as possible as shown in fig. (7), and then record them in the external contour database giving them a respective order.

### • Using IGES file

1. If the loop of external normal contour, which searches for (0, 0, Zmin) point, does not match this point in line or arc tables. A new loop will search in circles table for a center point has minimum X and minimum Y and minimum Z.
2. If this center point is found, that means there is an external circular contour.
3. The loop then turns into searching for a vertical line which starts or ends with  $(X_{center} + \text{CIRCLE radius})$ .
4. The depth of the circular contour is determined from the length of this line.
5. After all the external circular contours data is recorded in external table in database all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.
6. The software is able to recognize as many circular contours as possible as shown in fig. (8), and then record them in the external contour database giving them a respective order.

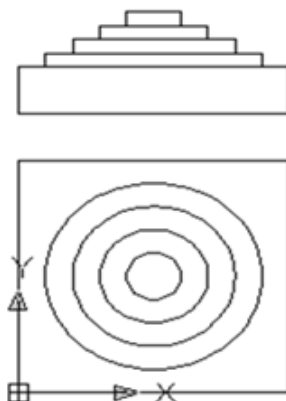


Figure (7) Circular external contours Using DXF file



Figure (8) Recognition of circular contours using IGES file



The output data from the recognition of this feature which can be used later are:

- Start and end points of this contour.
- The radius of this circular contour.

### Recognition of Internal Contours

- **Using DXF file**

The technique used to recognize these types of features is explained in the following steps:

1. The recognition loop searches the entire database for a line or an arc that starts with  $(X_{min}, Y_{min})$  point found on an entity (a line or an arc).
2. After finding  $(X_{min}, Y_{min})$  point the loop connects it with the next entity (line or arc) by its end point.
3. The loop keeps on searching for the ends for entities (lines or arcs) and link them with the starts of the next entities (lines or arc) until it reaches the starting point  $(X_{min}, Y_{min})$  again.
4. After the contour is recognized the depth of the contour is found by searching in contour data for the point  $(X_{min})$  and  $(X_{max})$  and then search in lines table in (X-Z) plane for the point corresponding to both as shown before in figure (9).
5. Before recording the contour data the software checks the line type in (X-Z) plane table in the database and there are two cases:
  - a. If the line type is dashed line: the recognition loop records these data in a table which will contain the data of internal contours, pockets and keyways which will be separated later.
  - b. If the line type is solid line: the recognition loop records the data in external table and this table contains as many data of external contours as possible (as explained later in 5.2).
6. After the entire contour data is recorded respectively in their tables in the database, all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.

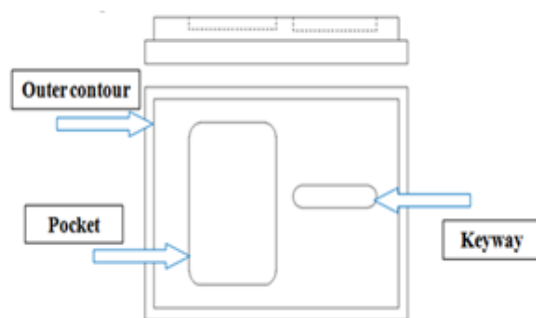


Figure (9) Recognition of internal contours using DXF file

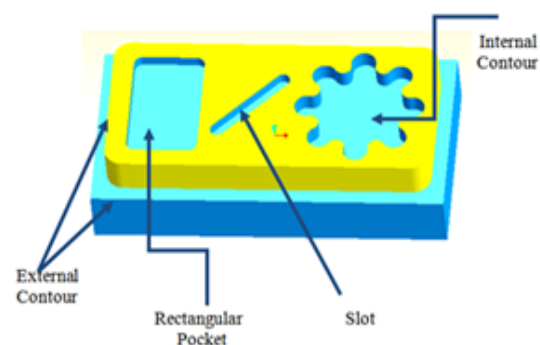


Figure (10) Recognition of internal contours using IGES file

- **Using IGES file**

The technique used to recognize these types of features is explained in the following steps:

1. The recognition loop searches the entire database for an entity (a line or an arc) that starts with a point having  $(X_{min}, Y_{min}, Z)$ .
2. After finding  $(X_{min}, Y_{min}, Z)$  point the loop connects it with the next entity (line or arc) by its end point.
3. The loop keeps on searching for the ends for entities and link them with the starts of the next entities until it reaches the starting point  $(X_{min}, Y_{min}, Z)$  again.
4. Before recording the data for this contour, the software searches the external contour table for an external contour having the Z value equal to Z value of the point of  $(X_{min}, Y_{min})$ .
5. If the  $(X_{min}, Y_{min})$  is greater than the minimum value of X and Y in all the elements of the founded external contour, this means that the following contour will be internal one.
6. After the internal contour is recognized, the depth of the contour is found by searching the vertical line table for a line starts or end with the point  $(X_{min}, Y_{min}, Z)$ . The depth of the internal contour is determined from the length of this line.
7. After all internal contours data is recorded respectively in their tables in the database, all the recorded data is deleted from the lines and arcs database to prevent the feature recognition module to be confused.

**The output data from the recognition of this feature which can be used later are:**

- The start and end points of each element of this internal contour.
- The type of each element.
- Determination of length of line and radius of arc of each element.
- The internal contour depth.

### **Recognition of Slots and Rectangular Pockets**

- **Using DXF file or IGES file**

From the previous explanation we notice that the data of any recognized feature will be deleted from the database after recording them in the corresponding table of features, so a new  $(X_{min}, Y_{min})$  points always will be exist deleting the data of any feature. The technique used to differentiation between the different types of internal contours (slots, rectangle pocket, or internal contour) as shown in figures 10, 11, 12 is explained as follows:

1. The software is searching the internal contour table for all elements of each contour and counts the number of database table's records.
2. If the number of records for any contour is 4 and satisfies two conditions, it will be recognized as a Slot. The first condition is the four elements should be in the following sequence: line, arc, line and arc and the radius of the two arcs are equal. The second condition is the difference between start and end angle of the two arcs is equal to 180 degree.
3. If the previous conditions are not satisfied then this contour with 4 elements will be recognized as internal contour.

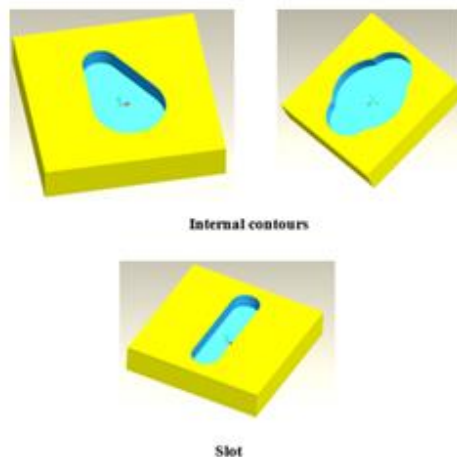


Figure (11) Recognition of slot

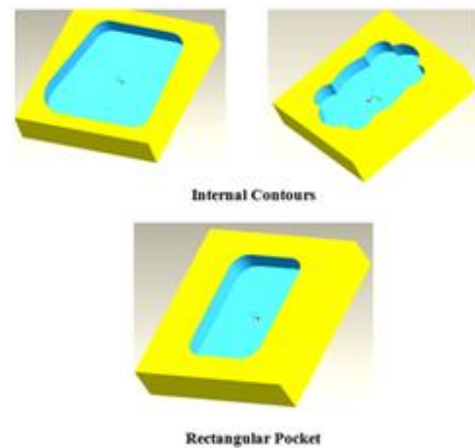


Figure (12) Recognition of rectangular pocket

4. If the number of records for any contour is 8 and satisfies two conditions, it will be recognized as a Rectangle pocket. The first condition is the eight elements should be in the following sequence: Line, arc, line, arc, line, arc, line and arc and the radius of the four arcs are equal. The second condition is the difference between start and end angle of the four arcs is equal to 90 degree.
5. If the previous conditions condition is not satisfied then this contour with 8 elements will be recognized as internal contour
6. If the number of element not equal 4 or 8 is recognized as internal contour.
7. The software is able to recognize as many pockets and slots as possible, and then when recording them in the database it gives them a respective order.

**The output data from the recognition of Rectangle pocket which can be used later are:**

- Start point.
- Pocket length.
- Pocket width.
- Pocket depth.
- Pocket corner radius.

**The output data from the recognition of slots which can be used later are:**

- Coordinate of any arc center.
- Length between two centers.
- Slot width.
- Slot radius.
- Slot depth.

### Recognition of Holes

- **Using DXF file**

1. Recognition loop searches in circle table in (X-Y) plane for all the circles and then record its diameter and  $(X_{center}, Y_{center})$  points.

2. The loop then turns into searching for the projection line of this circle in (X-Z) plane table having a point with X value equal to ( $X_{center} + \text{CIRCLE radius}$ ) and assure that this projection line is a dash line [ not continuous line]
3. The depth of the circler is determined from the length of the projection line as shown in figure (13).
4. Then the loop records the data of this circle and depth in the holes table.
5. After all the hole data is recorded in hole table database, all the recorded data is deleted from database to prevent the feature recognition module to be confused.
6. The software is able to recognize as many holes as possible.

• **Using IGES file**

1. Firstly we need to recognize a circle by using a smart loop search in arcs table for two arcs have the same center point and radius and the difference between start and end angles of these two arcs equal to 180 degree.
2. After the data is recorded in the circle table it will be deleted from arcs table to prevent software conflict.
3. A smart loop will be generated to search in circle table for two circles have same radius and values of X center and Y center and have different values in Z center.
4. Then this data will recorded in Hole table by using (X center, Y center, max. Z center1) as center of hole and the difference between Z center1 and Z center2 as hole depth figure (14).

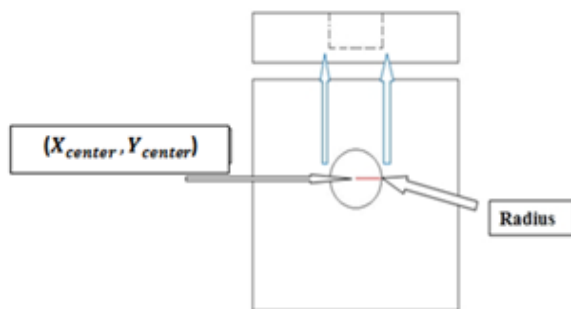


Figure (13) Recognition of holes Using DXF file

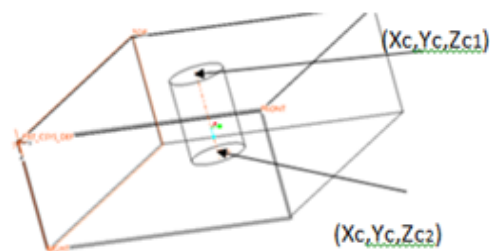


Figure (14) Recognition of holes Using IGES file

**Recognition of Flat and Taper Ended Holes**

• **Using DXF file**

Technique has been used to differentiate between flat and taper ended holes as shown in figure (15), (16). :

- A check is made in line table in (X-Z) plane for the depth of ( $X_{center}$ ) point , if that depth is equal to the depth of the point ( $X_{center} + \text{radius of circle}$ ) so the hole is a flat bottom hole and the hole type is recorded in the hole table.
- If the previous condition is not satisfies and the depth of ( $X_{center}$ ) point is not equal to the depth of the point ( $X_{center} + \text{radius of circle}$ ), the hole will be considered as taper ended hole and the hole type is recorded in the hole table in the field of hole type.

- If the hole is taper ended hole, a search will be made in (X-Z) line table for a vertical line has start point  $(X_C + \text{hole radius}, Z_C)$  and get its end point.
- The developed software searches for a line starts with the end point of the previous line and gets the end point of that line.
- The hole depth is then can be calculated by subtracting  $Z_C - Z_1$  as  $Z_1$  is the coordinate Z the of the founded inclined line as shown in fig (15).

• **Using IGES file**

- A search will be made in inclined lines table to find a line with one of the following conditions:
  - The line start with point  $(X_{\text{center}} + \text{hole radius}, Y_{\text{center}}, Z_{\text{center}2})$
  - or the line starts with a point  $(X_{\text{center}}, Y_{\text{center}} + \text{hole radius}, Z_{\text{center}2})$
  - or the line start with point  $(X_{\text{center}} - \text{hole radius}, Y_{\text{center}}, Z_{\text{center}2})$
  - or the line starts with a point  $(X_{\text{center}}, Y_{\text{center}} - \text{hole radius}, Z_{\text{center}2})$
- If a line of one of the previous condition is found, the software checks if that line ends with point  $(X_{\text{center}}, Y_{\text{center}}, \text{any } Z_{\text{value}})$ . If it satisfies the previous condition, so the hole is a taper ended hole and the hole type is recorded in the hole table.
- If software does not find this line, the hole will be considered as flat ended hole and the hole type is recorded in the hole table.

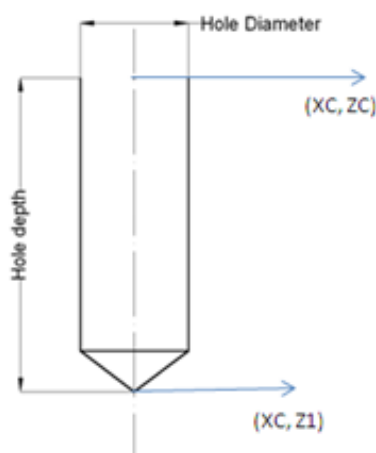


Figure (15) Recognition of taper hole Using DXF file

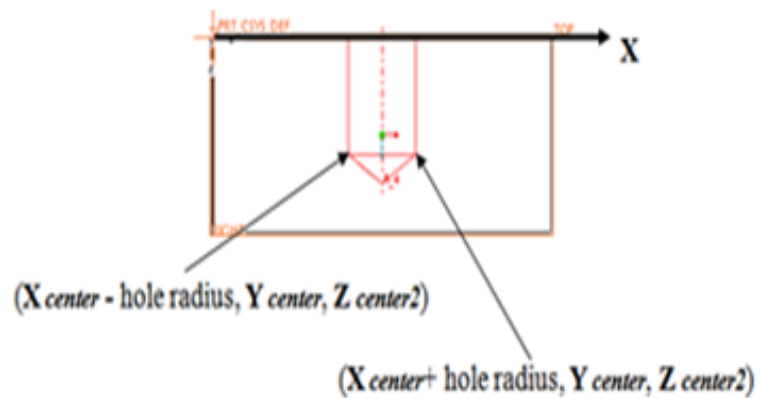


Figure (16) Recognition of Taper Hole Using IGES file

**Recognition of Threaded Holes**

• **Using DXF file**

1. Recognition loop searches in arc table in (X-Y) plane for all arcs that have same center points of any hole recorded in hole table and then record the arc diameter.
2. The loop then turns into searching for the projection line of this arc in (X-Z) plane table having a point with X value equal to  $(X_{\text{center}} + \text{arc radius})$  and assure that this projection line is a dash line [ not continuous line]
3. The depth of the thread is determined from the length of the projection line as shown in figure (17).
4. Then the loop update the data recorded for this hole to be threaded hole and save the thread diameter and thread depth.

5. After all the thread data is recorded in hole table database, all the recorded data is deleted from database to prevent the feature recognition module to be confused.
6. The software is able to recognize as many threaded holes as possible.

• **Using IGES file**

1. A smart loop will be generated to search in circle table and hole table for a circle has same center point of a hole and has different radius (R2).
2. Then software will search in circle table for another circle has the same values of X center, Y center, and same radius (R2) but has different value of its Z center (Z2).
3. Then the loop update the data recorded for this hole to be threaded hole and save the thread diameter equal to 2 XR2 and thread depth equal to (Z center – Z2) as shown in figure (18).

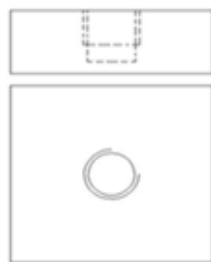
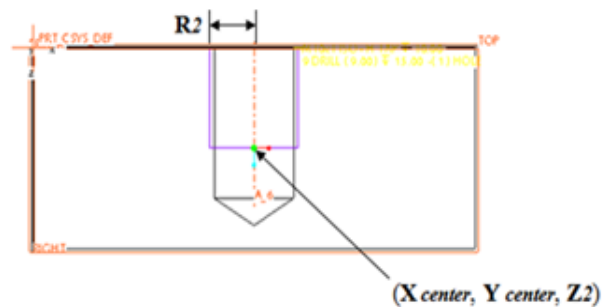


Figure (17) Recognition of threaded hole Using DXF file



(18) Recognition of threaded hole Using IGES file

The output data from the recognition of this feature which can be used later are:

- Hole center.
- Hole depth from its start plane.
- Thread diameter and pitch
- Depth of tapping on this hole.

**Recognition of Group of Holes on Pitch Circle**

• **Using DXF file**

The recognition module of holes on pitch circle has been developed in the following steps as shown in figure (19):

1. A search is done in circle table for all circles which drawn by center line type and get its radius  $R_p$  and its center point.
2. Using a smart loop to check that the distance between each hole center and pitch circle center equal to  $R_p$ . If this condition is satisfied for any hole, this means the hole will be included on pitch circle group.
3. Using another smart loop to get all holes which satisfies this condition and record the center point of each hole and calculate its angle by the following equations:
4. Hole angle =  $\tan^{-1}\{ (Y_{C \text{ hole}} - Y_{C \text{ pitch circle}}) / (X_{C \text{ hole}} - X_{C \text{ pitch circle}}) \}$

- All holes on pitch circle are recorded in another table and deleted from the old table of holes.

- Using IGES file**

The recognition module of holes on pitch circle has been developed in the following steps as shown in figure (20):

- All axes in the CAD model recorded in an hole axes table.
- A technique will be used to select all axes which pass through a circumference of hole and delete it from the previous table.
- A loop has been generated to find the distance between hole's centers and each remaining axis in axes table by using the following equation.  

$$\text{Distance} = \sqrt{\{ (Y_{C \text{ hole}} - Y_{C \text{ axis}})^2 + (X_{C \text{ hole}} - X_{C \text{ axis}})^2 \}}$$
- Using a smart loop to get all holes which satisfies this relation ( have same distance from axis) and record the center point of each hole and calculate its angle and pitch circle radius by the following equations:  
 Hole angle =  $\tan^{-1}\{ (Y_{C \text{ hole}} - Y_{C \text{ axis}}) / (X_{C \text{ hole}} - X_{C \text{ axis}}) \}$   
 Pitch circle radius = distance between hole centers and axis of the pitch circle
- All holes on pitch circle are recorded in another table and deleted from the old hole table.

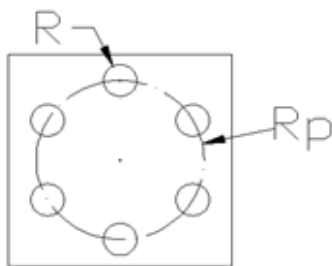


Figure (19) Recognition of group of holes on Pitch circle using DXF file

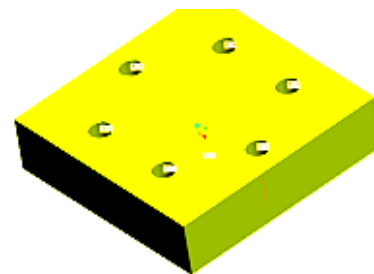


Figure (20) Recognition of group of holes on pitch circle using IGES file

**The output of recognition of this feature which required for the next modules are:**

- Center of pitch circle.
- Radius of pitch circle.
- Number of holes on this pitch circle.
- Angle of each hole.
- Start plane of holes.
- Hole depth.

### Recognition of Row of Holes

- Using DXF file and IGES file**

The row of holes can be defined as a group of holes having the same parameters (same diameter, thread, depth and end) and have at the same time a constant distance between the centre of any two successive holes (in X axis, or in Y axis or in both X and Y axes). Examples of row holes are shown in Fig (21) and Fig (22).

1. Firstly software searches in the holes table to find all holes which have same parameters (same diameter, thread, depth and end) and divide it into groups with similar parameters.
2. A smart loop has been developed to find the hole which have minimum coordinate value for X and Y axes of the hole center and name it as the main hole.
3. A loop has been generated to check if there is any relation between all holes in each group or not.
4. If distance in X direction equal to zero then we have a row of holes in Y direction. If distance in Y direction equal to zero then we have a row of holes in X direction. Else neither distance in X direction nor distance in Y direction equal to zero then we have an inclined row of holes.
5. The distance between the centers of each group of holes is calculated using the information in the database.
6. The data for each row of holes is recorded in the database with its parameters.

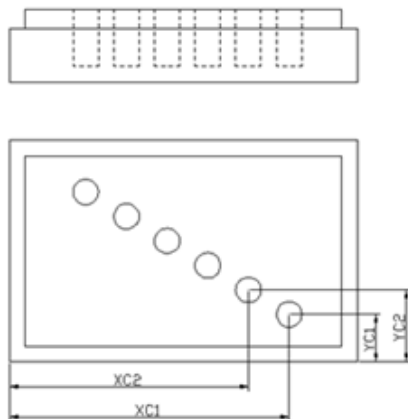


Figure (21) Recognition of row of holes Using DXF file

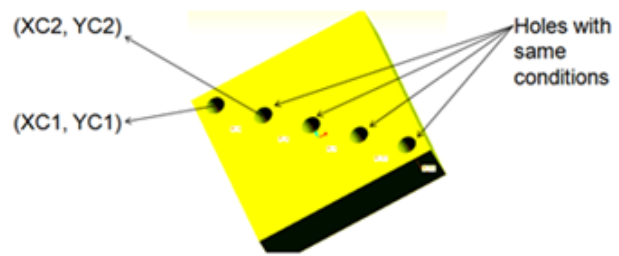


Figure (22) Recognition of row of holes Using IGES file

**The output of recognition of this feature which required for the next modules are:**

- First hole center.
- Distance between the centers of each two successive holes.
- Hole start plane.
- Hole depth.
- Hole radius.

### Recognition of Circular Pockets

- **Using DXF file or IGES file**

Another technique is used in hole recognition from the hole table. This technique has the following steps:

1. The circular pocket is recognized as a hole from the flat ended hole table.
2. The software compares the tools in the tools database with the diameter of the hole



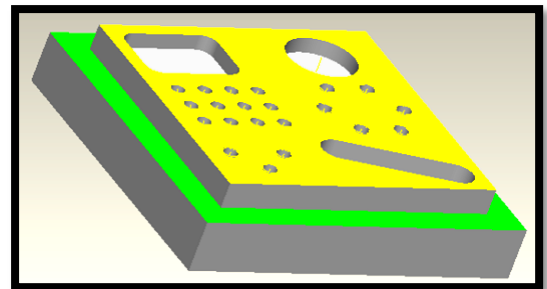
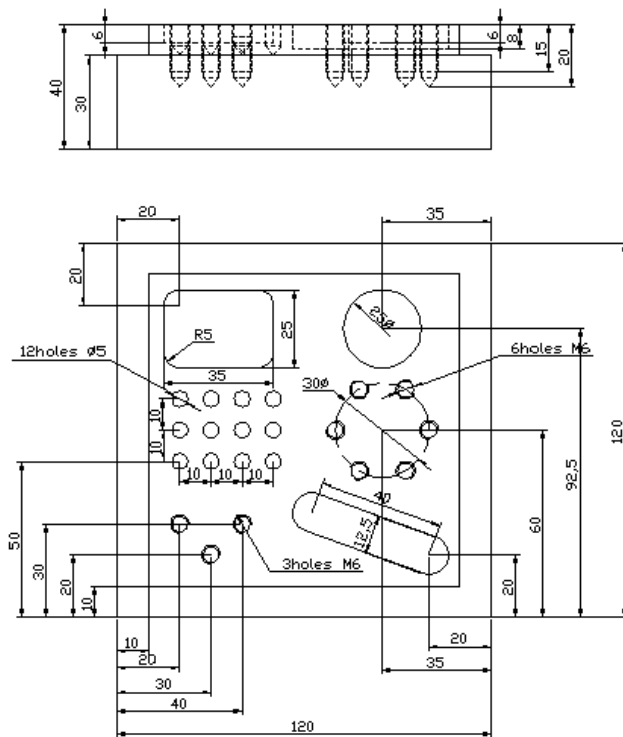
3. If the hole diameter is larger than the biggest diameter in the tools database, This hole will be considered as a circular pocket.
4. The software is able to recognize as many circular pockets as possible, and then when recording them in the database it gives them a respective order.

**The output data from the recognition of this feature which can be used later are:**

- Pocket center.
- Pocket radius.
- Pocket start plane.
- Pocket depth.

### CASE STUDY

A case study is designed to include as much features as possible to check the capability of the software to recognize these features with its required information.





## CONCLUSION

This paper includes a complete feature recognition technique for prismatic parts which is capable to deal with DXF file (in two views) and IGES file (as wire frame model) as an input data for any part drawing.

The developed software is also capable of the following:

- Extraction of the important lines, arcs, (circles in case of using DXF or axes in case of using IGES file).
- Preparation of the extracted data for creating manufacturing features.
- Joining the prepared data within certain logic to create manufacturing features.
- The developed software can deal with wide range of milling manufacturing features, 11 milling features.
- Calculating the size of the required raw material.

## REFERENCES

- [1] *D. Sreeramulu and C.S.P. Rao*, "A New Methodology for Recognition of Milling Features from STEP File" National Institute of Technology, Warangal – INDIA 2008.
- [2] *Emad S. Abouel Nasr and Ali K. Kamrani*, "A new methodology for extracting manufacturing features from CAD system" Industrial Engineering Department, Faculty of Engineering, University of Houston, USA, Available online 14 September 2006
- [3] *JungHyun Hana, MujinKangb, HoogonCho*, "STEP-based feature recognition for manufacturing cost optimization" School of Electrical and Computer Engineering, SungKyunKwan University, Suwon, 440-746, South Korea, (2001).
- [4] *Lian Ding, Yong Yue*, "Novel ANN-based feature recognition incorporating design by features" Department of Computing and Information Systems, University of Luton, Park Square, Luton LU1 3JU, UK, (2004).
- [5] *Mangesh P. Bhandarkar, RakeshNagi* "STEP-based feature extraction from STEP geometry for Agile Manufacturing Department of Industrial Engineering, 342 Bell Hall, State University of New York at Buffalo, Buffalo, NY 14260, USA, (1999).
- [6] *T. Derili, H. Filiz*"A note on the use of STEP for interfacing design to process planning" Department of industrial engineering, university of Gaziantep, Turkey (2002).
- [7] *Y. Woo', E. Wang', Y. S. Kim', H. M. Rho2*, "A Hybrid Feature Recognizer for Machining Process Planning Systems" School of Mechanical Engineering, Sungkyunkwan University, Suwon, Korea, (2004).
- [8] *Chee Fai Tan, 1V.K. Kher and 2N. Ismail*"Design of a Feature Recognition System for CAD/CAM Integration" World Applied Sciences Journal 21 (8): 1162-1166, 2013
- [9] *T. J. Jones1, C. Reidsema1 and A. Smith2*, "Automated Feature Recognition System for Supporting Conceptual Engineering Design", 2006
- [10] *A. Nassehia,\* , S.T. Newmanb, R.D. Allen*,"The application of multi-agent systems for STEP-NC computeraided process planning of prismatic components" Available online 15 August 2005.

- [11] *M. Kanga,\**, *J. Hanb*, *J.G. Moonc* "An approach for interlinking design and process planning" *Journal of Materials Processing Technology*, 2003.
- [12] *S.H. Suh\**, *B.E. Lee*, *D.H. Chung*, *S.U. Cheon*, "Architecture and implementation of a shop-floor programming system for STEP-compliant CNC" San 31 Hyoja-dong, Pohang 790-784, South Korea, (2003).
- [13] *Hyun Chan Lee \**, *Won ChulJhee*, *Hee-SokPark*"Generative CAPP through projective feature recognition" Available online 15 June (2007).
- [14] *Xionghui Zhou\_*, *YanjieQiu*, *GuangruHua*, *Huifeng Wang*, *XueyuRuan*"A feasible approach to the integration of CAD and CAPP A feasible approach to the integration of CAD and CAPP " National Die and Mold CAD Engineering Research Center, Shanghai Jiao Tong University,(2007).
- [15] *Jerry Y H Fuh*, *Chao-Hwa Chang\** and *Michel A Melkanoff* "The development of an integrated and intelligent CAD/CAPP/CAFP environment using logic-based reasoning", Department of Mechanical, Aerospace and Nuclear Engineering, UCLA, Los Angeles, CA 90024, USA (1995).
- [16] *George Vosniakos\**"An intelligent software system for the automatic generation of NC programs from wireframe models of 2-1/2D mechanical parts" Department of Mechanical Engineering, Manufacturing Division, National Technical university of Athens, Greece (1998).
- [17] *JIAN (JOHN) DONG 1*, *HAMID R. PARSAEI 2* and *HERMAN R. LEEP* "MANUFACTURING PROCESS PLANNING IN A CONCURRENT DESIGN AND MANUFACTURING ENVIRONMENT" Department of Industrial Engineering, University of Louisville, Louisville, KY 40292, U.S.A. (1996).
- [18] *A jay Joneja*"Geometric reasoning for optimizing backward growing-based feature recognition" Dept of Industrial Engineering & Engineering Management Hong Kong University of Science & Technology, (1997).
- [19] *Tienne Fortin*, *Jean-FrançoisChatelain\**, *Louis Rivest*"An innovative software architecture to improve information flow from CAM to " Available online 19 June (2004).
- [20] *Shaw C. Feng\**, *Keith A. Stouffer*, *Kevin K. Jurrens*"Manufacturing planning and predictive process model integration using software agents" Manufacturing Engineering Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA, (2005).