Jurnal Mekanikal
June 2012, No 34, 83-94

EFFICIENT VERTEX CLASSIFICATION METHOD TO RECOGNISE THE ORTHOGONAL AND NON-ORTHOGONAL PRISMATIC FEATURES

Gabriel Yap Wei Phing, Jamaludin Mohd Taib* and Masine Md Tap

Faculty of Mechanical Engineering,
Universiti Teknologi Malaysia,
81310 Skudai, Johor Bahru

ABSTRACT

The geometrical property of the topology becomes more specific as the topology goes down to lower level. Hence, it will have less information to handle and eventually accelerate the search process. This paper proposes a vertex classification method, which is the lowest topology data in boundary representation (B-Reps) model. Apart from having only one geometrical property to accelerate the search process, the iterative search process to classify the feature will further make the search process more effective. Firstly, identification of feature originated from vertices classified as Vertex Inside Stock (VIS) is carried out. When all the features from VIS are identified, the search for features emanated from vertices called Vertex On Stock (VOS) are commenced. By examining the adjacent vertices of the VIS or VOS, blind pocket, notch and side pocket are identified when the system examined VIS, whilst the presence of VOS will lead to the identification of slot and step. Finally, the evaluation of the algorithm shows significant improvement in recognition time compared to pattern matching algorithm.

Keywords : *Feature recognition, CAD, CAM, process planning, modelling.*

1.0 INTRODUCTION

Currently, Boundary Representation (B-Reps) is the most common solid geometric modeling available commercially. Hierarchical information of topology and geometry as its low-level description of a product is the basic of B-Reps data. In fact, topology and geometry form complicated network of data that demands the feature recognition system to rigorously examine all the data for its downstream activities. This in fact poses a barrier to feature recognition which researchers seek to overcome to ensure the efficiency of the search process.

This paper will propose a vertex classification method to make the recognition technique to be less dependent on the complicated network of information. Vertex classification is the foundation of the search process which allows a feature recognition system developed to focus on the search process from the bottom level of topology which is the vertex and its adjacent vertices. Based on the classification of vertices, features are defined. Moreover, the search process focuses on the topology that enables the orthogonal and non-orthogonal features to be identified in similar manner.

Section 2 explains the previous study of feature recognition system. Then, the overall procedure of the feature recognition is discussed in Section 3. Illustrative examples are demonstrated in Section 4. Discussion on the advantages of the vertex

*Corresponding author : jamalt@fkm.utm.my

classification method is explained in Section 5. Finally, the conclusion is drawn in Section 6.

2.0 LITERATURE REVIEW

In most of the previous work, the search for features on B-Reps model commences from the top level of topology. Amongst the earliest work, Joshi and Chang [1] used face as the foundation of the search. The faces are represented as the nodes of the Attributed-Adjacency Graph (AAG) representation of the model. Concavity between two faces is attached to the arcs of the graph. AAG has provided foundation for other research. Ibrahim and McCormack [2] used AAG and integrate it with hint-based taxonomy to identify more features.

In 1996, a three level graph based approach with hint used to decompose features is proposed by Rahmani and Arezoo [3]. Firstly, AAG is decomposed into Implicit Concave Graphs (ICG) based on the concave edges. ICG is then decomposed into final graph which is the graph that represents the features.

Apart from graph which can be directly translated from B-Reps data, feature coding has also been introduced. Venuvinod and Yuen [4] used feature coding, comprises of the feature properties and its faces, to recognize the features. String matching process will identify the features. Pattern matching using the position of point on the model has been proposed by Ismail et. al [5]. Later in 2008, Arivazhagan et al [6] extended the work on pattern matching to include the tapered and curved base features. Syntactic pattern matching representing the edges, loops and faces was adopted.

Some researchers use Artificial Neural Network (ANN) to identify features. However, this approach relies on the templates that have been used in the training section. With ANN, likeliness of features with feature template has been suggested by the system. Amongst the researchers are Meeran and Zulkifli [7]. They used templates in the form of matrices of 2D pattern consisting of edge and vertices using cross-sectional method. In 2006, binary representation of the edges and vertices as the input for ANN system was proposed by Chakraborty and Basu [8]. The length of the representation depends on the features. Nine bit binary is used for slot and step, whilst nine bit binary is required for circular features. In 2009, another researcher who adopted ANN in detection of features is Sunil and Pande [9]. They employed 12 knot vector scheme that represents the relationship of face, vertex, and edge, while in 2010, Guan et al [10] used ANN to identify feature from STEP data. Due to the nature of STEP data, lexical method was adopted and AAG is used to define the feature.

Differing from the previous work that uses template from the topology data, Ranjan et. al [11] proposed 2D feature pattern based on the 'ray tracing' method. A ray is shot from specified virtual face until the ray hit the face on the part. The length of ray between the virtual face and the point when the ray hit the face of the part creates the pattern for feature recognition system.

Most of the previous researches use the higher hierarchy of the topology data. As far as the topology data is concern, searching higher or lower level of topology data has no effect on the efficiency of the search process. However, retrieving the geometrical data for respective topology may affect the search process.

The nature of the topology and geometry data is that the top level topology data has various geometrical types with each type may have unique characteristics and as the topology gets to the lower level, the geometry becomes more specific. As such, face, which is considered at the top level topology data, comprises a number of surfaces, such as plane, cylindrical, conical or spherical. Each surface will have its unique geometrical information such as radius of the circular arc for cylindrical surface, and plane which do not need radius as one of its geometrical information. As the hierarchy level goes down the topology data, its geometrical data becomes specific, to the extent that at the lowest topology data, which is vertex, it will have only one geometrical data, that is, coordinate of the vertex.

Due to the behavior of the topology and geometry data, the search process becomes more efficient when the search starts from the lower level of the topology data. Therefore, this paper is proposed a vertex classification method to identify isolated features. Vertices will be classified as Vertex Inside Stock (VIS) or Vertex Outside Stock (VOS) based on the position of the vertices with respect to the stock.

The technique of defining the features has the effect on the efficiency of the search process. Pattern matching as previously discussed poses computational complexity as each feature must be cross examined with all the available templates. Therefore, in the system presented in this paper, the presence of features will be hinted by the presence of VIS and VOS, followed up with confirmation process to define the feature. This study is part of a larger system. The features from inner loops are excluded as it has been discussed in Zhen et al [12] in 2012. Gindy's taxonomy [13] is adopted in the study to define the features.

3.0 FEATURE RECOGNITION APPROACH

This section elaborates the feature recognition procedure adopted by the system, which is illustrated in Figure 1. In general, feature recognition procedure is divided into two-stage; gathering preliminary data, then followed by Vertex Classification Method. Section 3.1, gathering preliminary data will examine the input as well as retrieve the stock size for process planning purposes.

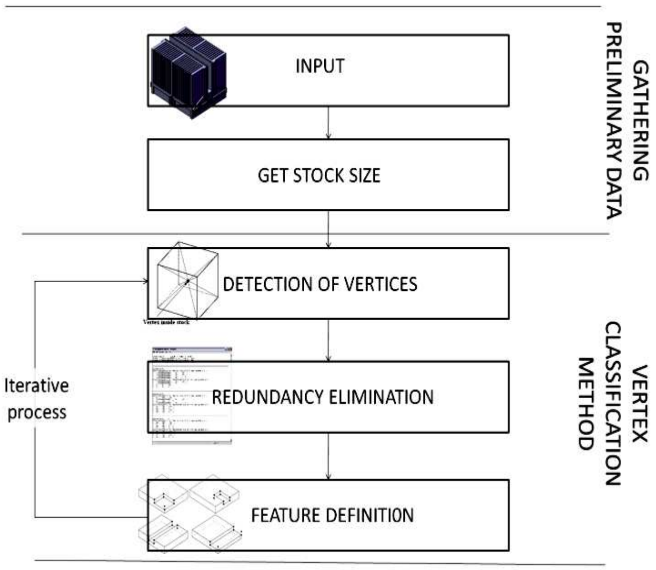


Figure 1 : Feature recognition approach

Section 3.2 will discuss the core of the feature recognition, which is the Vertex Classification method. The Vertex Classification method starts with detection of the vertices, which is called Vertex Inside Stock (VIS) and Vertex Outside Stock (VOS). Then, VIS and VOS will be verified with previous features to check if they have been identified and this stage is called redundancy elimination stage. Finally, features are then defined based on connectivity between VIS and VOS. The whole vertex classification method is carried out based on a single vertex, and then it is carried out iteratively until all the vertices are examined.

3.1 Gathering Preliminary Data

This stage will gather the preliminary information related to process planning. The stock size is the major parameter for process planning. However, prior to retrieve the stock size, the system will verify whether the model has been created or not. If there is no created model, the system will exit and message the user to remind the user to create the part.

In the case of the part has been created, the system will visit all the faces, and the co-ordinates of the vertices will be compared with minimum and maximum values for each axis to get the boundary of the part in space. Finally, the length, height, and width are calculated based on the distance between the minimum and maximum value on X, Y, and Z axis respectively.

3.2 Vertex Classification Method

Vertex classification method is the core for the feature recognition technique, which comprises of three-step recognition process; vertex detection, redundancy elimination, and feature definition. This three-step recognition process is carried iteratively on each vertex. Once the vertex is detected, the system will verify whether the detected vertex has never been examined and finally, the feature emanated from the vertex will be identified, prior to proceeding to another vertex with the same three-step process.

3.2.1 Vertex detection

Vertex detection stage will search for vertices and then classify them as either Vertex Inside Stock (VIS) or Vertex On Stock (VOS). Vertex that is positioned inside the stock is called VIS as shown in Figure 2a, whilst when the vertex is at inner position of the stock outer faces is called VOS as shown in Figure 2b.

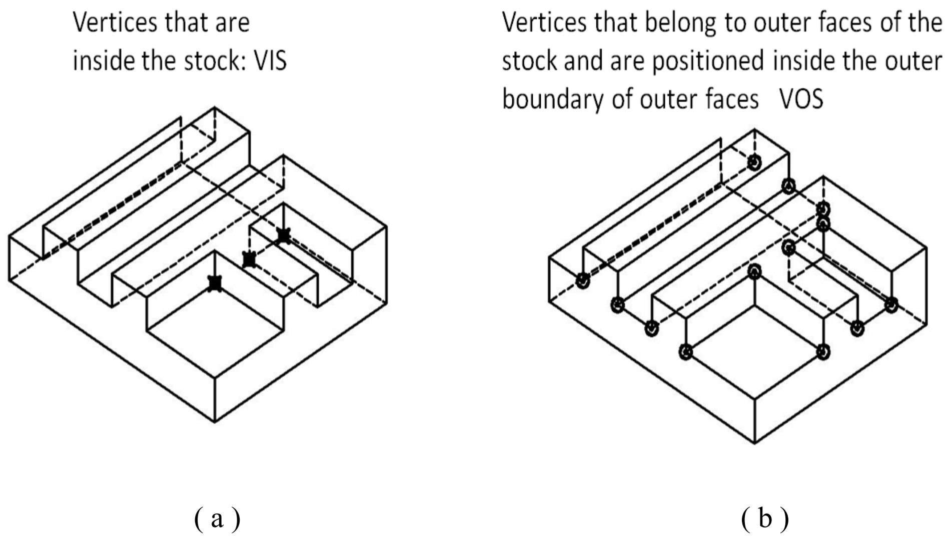


Figure 2 : (a) Vertex inside stock (VIS) and (b) vertex outside stock (VOS)

In the case of searching of VOS vertices, the search is only limited to faces that form the outer faces of stock with more than four vertices. This is because when features exist on the faces, the features will cut the face and this causes the vertices on the faces to have more than four vertices. Moreover, the condition will also eliminate the possibilities of detecting the same vertex from different faces. This is because every vertex is at least the incidence of three faces. However, no condition is applied when searching the VIS, any vertex that is inside the outermost face will be classified as VIS.

3.2.2 Redundancy Elimination

Some of the features may have more than one vertex that is VOS or VIS or both of them. Having more than one vertex that are VOS or VIS may result in identifying more than one features from the same features. Therefore this stage will discard the VIS or VOS that belongs to the previous identified features.

3.2.3 Feature Definition

After VOS and VIS have been identified in previous stage, features that are emanated from these vertices should be identified. Dual-level comformation method is adopted here. Every vertices will have at least three incidence edges and each edges will have two vertices; based on the connectivity between the vertices, features can be defined.

The search commences from the VIS. The possible features that have VIS are notch and side pocket. The difference between notch and side pocket is that the VIS in the side pocket is connected to another VIS vertex, whilst VIS vertex of the notch is not. Therefore, when VIS is connected to another VIS, side pocket is detected; and when VIS is not connected to another VIS, notch is identified.

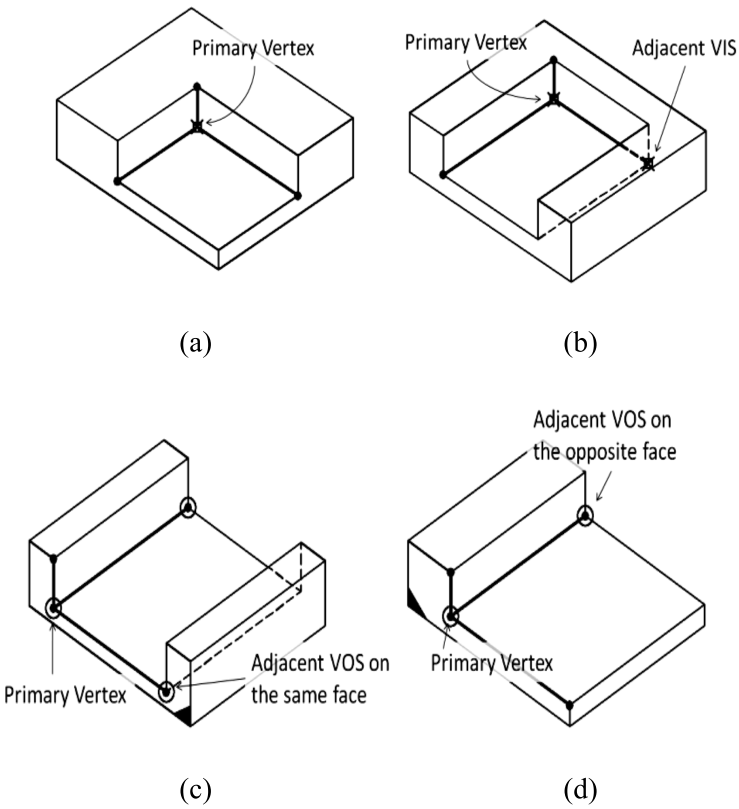


Figure 3 : Feature definition based on vertices

After the features from VIS are defined, the next step is to define features that originated from VOS. Two features can be detected here, namely; step and slot. Slot is detected when the VOS vertex is connected to another VOS vertex that is on the same outer face. When VOS is connected to another VOS vertex, but on the different outer face, step is detected. Table 1 summarizes the attributes of the features and Figure 3 illustrates these feature definition.

Table 1 : Attributes of Selected Features

	Primary Vertex	Adjacent Vertices
Notch	VIS	Not connected to another VIS
Side Pocket	VIS	Connected to another VIS
Slot	VOS	VOS on the same face
Step	VOS	Not connected to any VOS on the same face

3.2.4 Searching The Vertices of The Features

This stage will establish the volume of the features. Based on the removal of the material during machining, it will leave behind the bottom face of the features, therefore this bottom face can be used as the starting point to search for the volume of the features. To identify the bottom face, the following unique characteristics for each feature can be used:

Notch : primary vertex and any two of adjacent vertices
Side pocket : primary vertex, adjacent vertex which is VIS, and one of the remaining vertices
Slot : primary vertex and two of the vertices that are VOS
Step pocket : primary vertex , adjacent vertex which is VOS, and one of the remaining vertices

The next stage is to identify the vertices of the top face of the feature. Since top face of the features are being cut during the removal of the volume during machining, direct search process using the topology data is not an option to be considered. Therefore, the search will be based on the vertices of the bottom faces and geometrical reasoning on these vertices will identify the vertices and even the additional vertex/vertices required.

Due to characteristics of the features, the geometrical reasoning for the features is as follows;

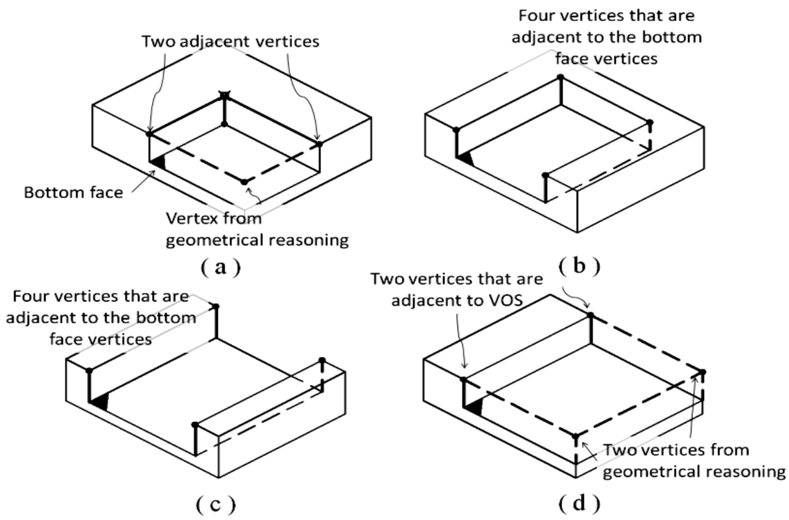


Figure 4 : Defining top face vertices

Notch : The adjacent vertex that is not one of the vertices of the bottom face will be used as the reference to search the vertices of the top face. Similar method to identify the adjacent vertices will be used to identify the other two vertices of the top face. By excluding the primary vertex from the list, the remaining vertices are the vertices that form the top face. The additional vertex will be based on the geometrical reasoning of the three vertices identified. Since the additional vertex must be one of the corners of the stock, geometrical reasoning can be applied to calculate the vertex (refer to Figure 4a).

Side pockets : Vertices of the top face are in fact the adjacent vertices to the bottom face. Therefore, the search of the adjacent vertices of the bottom face vertices will lead to identifying the vertices of the top face (refer to Figure 4b).

Slot : Similar method to vertices of the top face of side pocket is applied here (refer to Figure 4c).

Step : Only two of the vertices can be identified and the other two vertices have to be calculated. Based on VOS vertices, the two vertices can be identified by searching their adjacent vertices. The other two can be calculated using geometrical reasoning based on the observation that the step must be present on one of the edges of the stock (Refer to Figure 4d).

The search of the vertices of the top and bottom faces is based on the topology. The main advantage of the topological based searching method is that the system will be able to identify the features regardless of their orientations. Therefore, both orthogonal and non orthogonal features can be identified using the same procedure. Sample of the non-orthogonal features is shown in Figure 5.

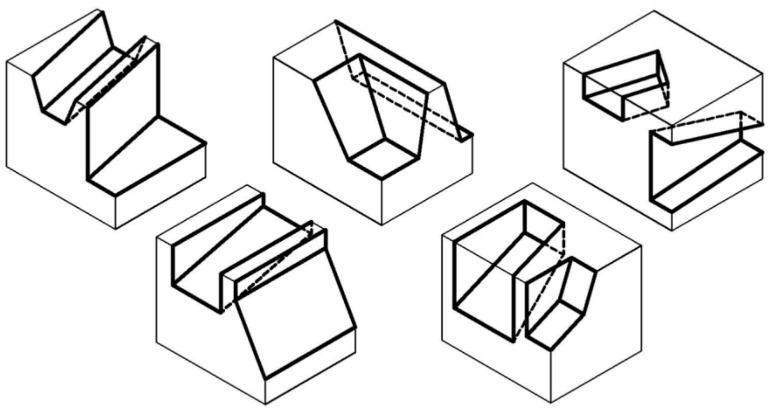


Figure 5 : Non-orthogonal side and bottom face features

3.3 Programming Implementation

Figure 6 shows the programming implementation. Firstly, the system will retrieve the stock size and whilst retrieving the stock size, few parameters are saved. The parameters are the minimum and maximum value of x, y, and z coordinate and the list of the outer faces that are saved into list of faces according to the principal plane. These parameters will be used later for the purpose of programming.

The system proceeds with the identification of VIS and its features. VIS is identified when the coordinates of the vertex lies in between the minimum and maximum x, y, and z coordinates. Once VIS is detected, the system will verify whether the current VIS has never been visited at the redundancy elimination process. The system will proceed to feature identification process on the verified VIS only.

When all the features emanated from VIS are identified, the next stage is to identify the features originated from VOS. This is done by examining the outer faces saved during getting the stock size stage. The outer faces with more than four vertices have the potential to have features. However, there is possibility that VOS candidate belongs the feature that has been identified. Therefore, every identified VOS must go through the vertex redundancy elimination process. With the redundancy elimination process is carried out, the identified VOS will proceed to feature identification process. Finally, the process ends when all the outer faces are examined.

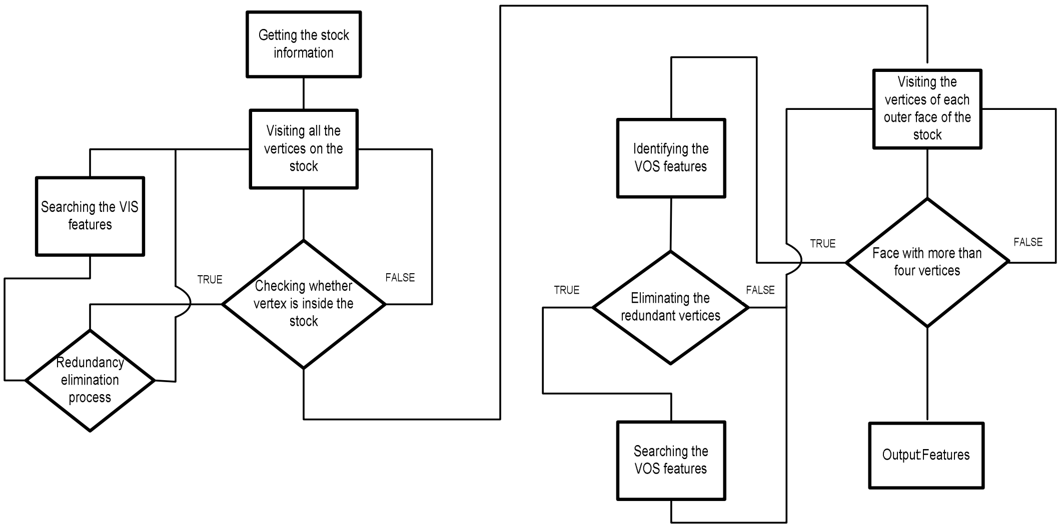


Figure 6 : Programming Implementation

4.0 ILLUSTRATIVE EXAMPLES

Figure 7 shows Part A as the first illustrative example, comprising of four different orthogonal features; slot, step, notch, and open pocket. When this model is used as the input to the system, the first step is to retrieve the size of the stock, and as a result, stock of 100 mm x 75 mm x 50 mm is identified.

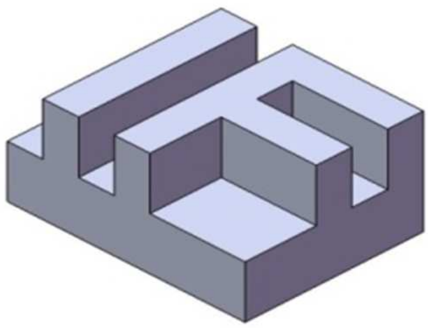


Figure 7 : Illustrative Example: Part A

Then, the system will carry out the vertex classification method to identify the features. Firstly, the system will identify the VIS vertices. The first VIS identified is *Vis1*. When the system searches the adjacent vertices, another VIS, which is *Vis2* is detected, and therefore, side pocket is identified. Then, the system proceeds with identify the volume of the side pocket and additional vertices, *V1* to *V6* are identified. After identification of the side pockets, two VISs (*Vis1* and *Vis2*) and six vertices (*V1* to *V6*) are identified as shown in Figure 8a. These vertices will be used during the redundancy elimination process.

When volume of the side pocket is identified, the system will proceed to identify more VIS. Here *Vis2* is identified, however it is removed during the redundancy process as it belongs to one of the side pocket vertices. Then, *Vis3* is identified and this vertex proceeds with feature identification. Since *Vis3* is not connected to any other VIS, notch is identified. When the volume of the notch is established, *V7* to *V13* are identified as shown in Figure 8b. *V13* is an additional vertex. Then the system searches for another VIS, however no more VIS is detected and this marks the end of searching process to identify feature emanated from VIS.

Then, the system proceeds to identify the features originated from VOS. First face *Fac1* is identified and on this face, 3 VOSs (*Vos1, Vos2, and Vos3*) are detected as shown in Figure 8c. However, these vertices are eliminated during the redundancy stage as they are belonged to other features.

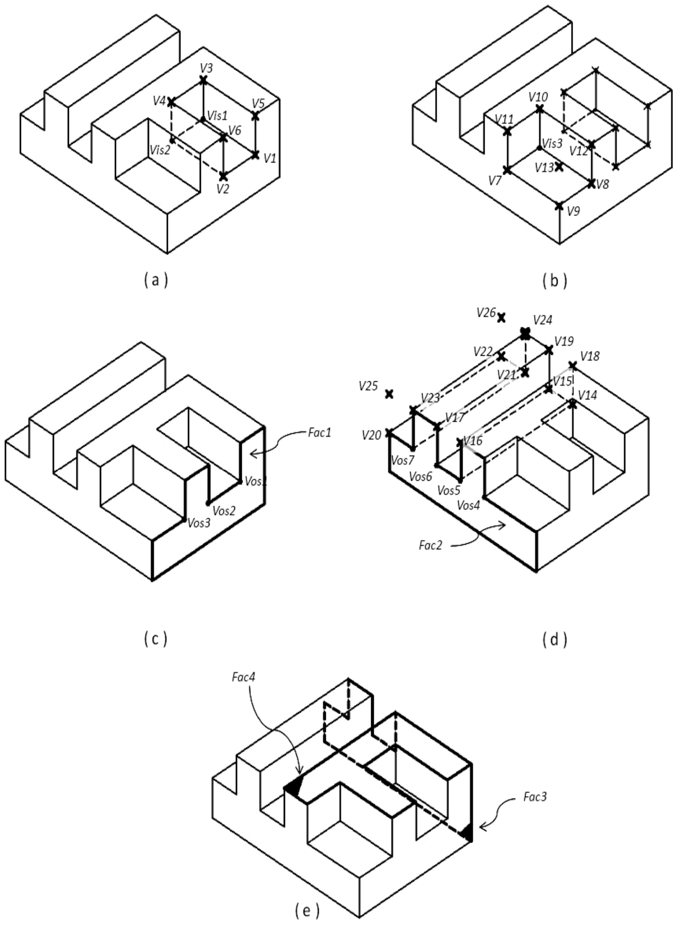
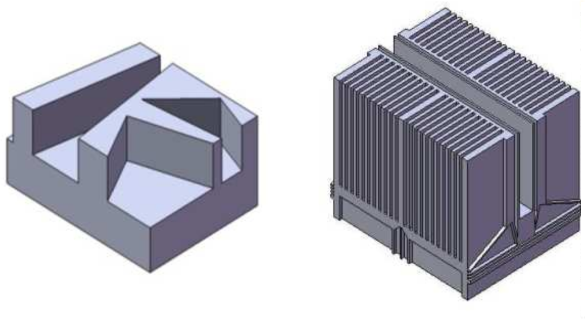


Figure 8 : Procedure to recognize the features

When the system searches for VIS on the next face (*Fac2*), four VOSs (*Vos4* to *Vos7*) are identified. *Vos4* is removed as it belongs to other feature. *Vos5* to *Vos6* will be as the primary vertex during feature recognition stage. When the system examines the adjacent vertices of *Vos4*, it discovers that *Vos4* is connected to another VOS, which is *Vos5*, on the same face. The connectivity of two VOS shows the presence of slot. Then, the system establishes the volume of the slot, and then vertices *V14* to *V19* are identified. Then the system proceeds with *Vos5*, however it is eliminated because it belongs to previously identified slot. However, when *Vos6* is examined, it is discovered that it is not connected to any other VOS on the same face, therefore, step is identified. When the volume of the step is established, *V20* to *V26*, as shown in Figure 8d, are identified. *V25* and *V26* are additional vertices.

Similarly the system will search VOS for other faces; only two faces (*Fac3* and *Fac4*) are the faces that have the potential VOS as shown in Figure 8e. However, all the identified VOSs are removed as they belong to other features.

Finally one side pocket, one notch, one slot, and one step are detected within 15.625ms. Similarly the approach can also identify non-orthogonal features as in Figure 9a. The system is also tested on more complex model as in Figure 9b. This model was created with a 100 mm x 80 mm x 80 mm stock and it comprises of the combination of 4 notches, 8 slots, 2 steps and 56 open pockets (a total of 70 features). There are also combinations of orthogonal and non-orthogonal features in this model with varying sizes. The total time for the algorithm to recognize these features is 217.7734 ms.



(a) (b)

Figure 9 : Tested models

5.0 DISCUSSION

A system based vertex classification method has been developed. Performance of the recognition technique has been tested by taking the time to execute when the number of features increases. To maintain the consistency of the evaluation, a 500 mm x 300 mm x 100 mm stock is used and the dimensions of the features are kept the same; step (100 mm x 300 mm x 40 mm), notch (100 mm x 100 mm x 40 mm), slot (5 mm x 300 mm x 40 mm), and open pocket (10 mm x 100 mm x 40 mm). For every knot, five readings are taken and the average is plotted. Figure 10 shows the graph of the number of features and the time taken to recognize them.

The development of this system has the following advantages that make the search process efficient:

i. Generic approach to identify the orthogonal and non-orthogonal features

Vertex Classification Method focuses on the search on topology and as a result, the orthogonal and non-orthogonal features will be identified in similar manner. Geometrical properties are used in the reasoning to define the vertex as well as the feature definition.

ii. Minimal topological examination

This method focuses on the classification of vertex. In the search of VIS, vertex is directly examined, whilst in the search of VOS, the system only examines the outer boundary faces only. This minimal topological examination makes the search process efficient.

iii. Classification of features

Classification of features demands the search process to be carried on VIS and then follows with VOS. Each detected vertex will have to go through a complete recognition procedure and then followed with another vertex. Cross examining the vertices of the features together with the VIS or VOS, the system will eliminate the possibility of having to visit the same vertices.

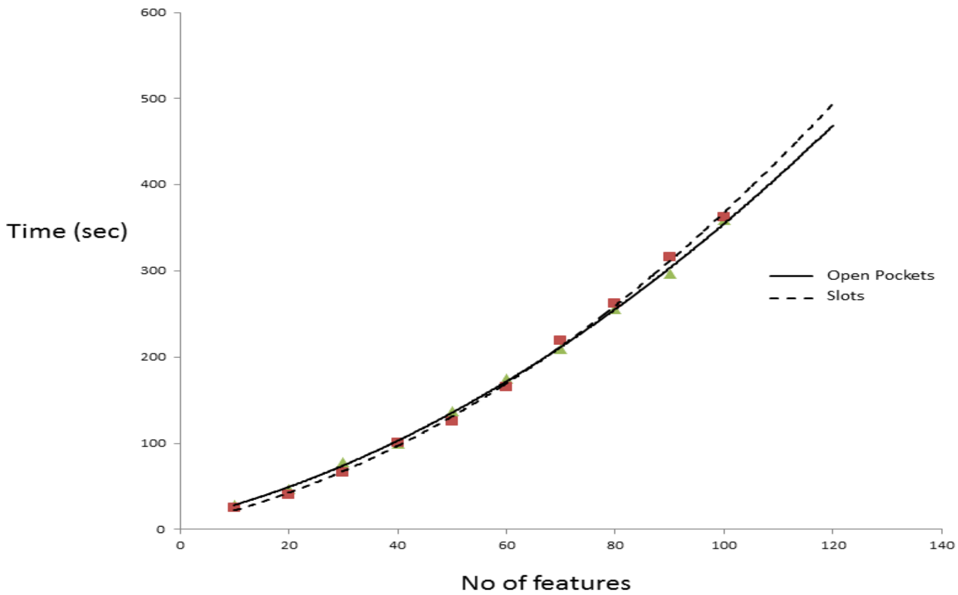


Figure 10 : Number of features vs. average time graph

6.0 CONCLUSION AND FUTURE WORKS

The algorithm to recognize a specific chosen set of features was successfully developed and the evaluation of the performance of the system has shown promising results. The features recognized by this system are in the form of machining features which have more emphasis on downstream activities. The algorithm has also successfully recognized features with tapered surface and non-orthogonal features. The recognition time for this system was recorded as a performance indicator for the efficiency of the system and can be used as a comparison with future works.

This system has its setback. The system cannot identify the non-orthogonal features in Figure 11.

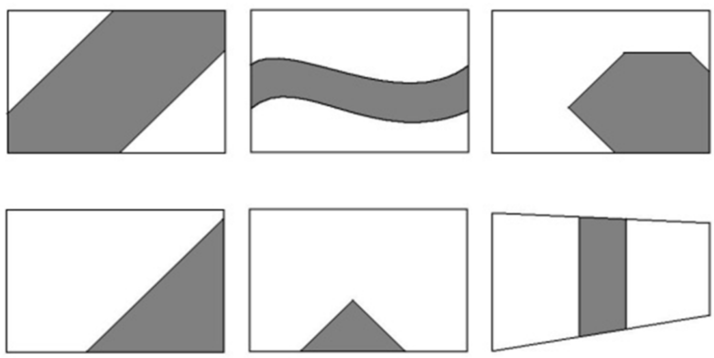


Figure 11 : Examples of invalid non-orthogonal features of the system

REFERENCES

1. Joshi, S. and Chang, T.C. "Graph Based Heuristics for recognition of machine features from a 3D solid model", Computer Aided Design, 20 (2), pp 58-66 (1988)
2. Ibrahim, R.N. and McCormack, A.D. "Robustness and generality issues of feature recognition for CNC machining", Inter J Adv Manuf Technol., 25 (7-8), pp 705–713 (2005)
3. Rahmani, K. and Arezoo, B. "Boundary analysis and geometric completion for recognition of interacting machining features", Computer-Aided Design, 38(8), pp 845–856 (2006)
4. Venuvinod, P.K. and Yuen, C.F. "Efficient automated geometric feature recognition through feature coding", Annals of CIRP, 43(1), pp 413-416 (1994)
5. Ismail, N. Abu Bakar, N. and Juri, A.H. "Feature Recognition Patterns for Form Features Using Boundary Representation Models", Int J Adv Manuf Technol., 20(8), pp 553–556 (2002)
6. Arivazhagan, A. Mehta, N.K. and Jain P.K. (2008) "Development of a feature recognition module for tapered and curved base features", Int J Adv Manuf Technol., 39(3), pp 319–332 (2008).
7. Meeran, S. and Zulkifli, A.H. "Recognition of simple and complex interacting non-orthogonal features", Pattern Recognition, 35(11), pp 2341 – 2353 (2002)
8. Chakraborty, S. and Basu, A. "Retrieval of machining information from feature patterns using artificial neural networks", Int J Adv Manuf Technol., 27(7-8), pp 781–787 (2006).
9. Sunil, V.B. and Pande, S.S. "Automatic recognition of machining features using artificial neural networks", Int J Adv Manuf Technol., 41(9-10), pp 932–947 (2009)
10. Guan, X. Meng, G. and Yuan, X. "Machining feature recognition of part from STEP file based on ANN" 2010 Int Conf on Computer, Mechatronics, Control and Electronic Engineering (CMCE), Changchun, China, pp 54-57 (2010).
11. Ranjan, R. Kumar, N. Pandey, R.K. and Tiwari, M.K. "Automatic recognition of machining features from a solid model using the 2D feature pattern", Int J Adv Manuf Technol, 26 (7-8), pp 861–869 (2005)
12. Zhen, Y. Mohd Taib, J. and Md Tap, M. "Decomposition of interacting machining features based on the reasoning on the design features", Int J Adv Manuf Technol, 58(1-4), pp 359–377 (2012)
13. Gindy, N.N.Z. "A hierarchical structure for form features", Int J of Prod Res., 27(12), pp 2089-2103 (1989).