Complex Hole Recognition from CAD Mesh Models

Vaibhav J Hase[1],Yogesh J Bhalerao[2], Saurabh Verma[3],
Sandip Jadhav[3],G JVikhe Patil[4]

[1]Research Scholar, SPPU, Department of Mechanical Engineering, AVCOE, Sangamner (India).
[2] Professor in Mechanical Engineering, MIT Academy of Engineering, Alandi, Pune (India).
[3]Centre for Computational Technologies, Pune (India).
[4] SPPU, Department of Mechanical Engineering, AVCOE, Sangamner (India).

ABSTRACT

This paper reports an algorithm to recognize complexintersecting holes from CAD mesh models based on hybrid mesh segmentation. The algorithm involves three steps viz. preprocessing, hybrid mesh segmentation andhole recognition. Inthe preprocessing step, we build a topology of the imported CAD mesh model. In the Hybrid Mesh Segmentation step, we cluster facets into groups based on mesh attributes. The facets in the clusters are then subjected to several conformal tests, to identify the type of analytical surface it might be representing, such as aplane, cylinder, cone, torus orsphere. In this research, arule-based approach is usedfor compound hole detection along with hole chains. This algorithmhas been implemented in VC^{++} and has been extensively tested on models taken from NIST repository for complex intersecting holes. The innovation lies in complexintersectinghole detection and parameterization.The proposed approach outperforms the existing techniques favorably and is found to be robust and consistent. This extracted feature information can be utilized during all stages of the design-to-manufacturing cycle.

Keywords:CAD mesh model, Compound hole recognition, hybrid mesh segmentation, intersecting Hole recognition.

1.Introduction

Compound hole features are ubiquitous in mechanical engineering applications from design to manufacturing cycle. In many mechanical engineering parts, compound holesconstitute a significant percentage of features. Recognizing compound holes in CAD mesh models is vital in applications such as mesh simplification, design, manufacturing, and FEA. Design engineers may be interested in modifying some hole parameters whereas FEA engineer may prefer to suppress some holesif these holes are not expected to contribute significantly during FEA.

Mesh models constructed from 3D scan data are called scan derived mesh and those generated from B-rep models using CAD software are called CAD mesh models (CMM). The focus of this paper is the CAD mesh model.

Segmentation aims to partition CMM into ‘‘meaningful’’ regions[1].Each region can be fitted to a distinct, mathematically analyzable form[2]. Literature reveals the availability of many mesh segmentation algorithms. However, most of them are not suitable for CMM. Several mesh segmentation approachesin the literature have relied on information such as curvature or sharpedges. Huge time is needed for curvature computation. The curvature is sensitive to noise, variations in dimensions and unevenly distributed triangulations [2]. Several mesh segmentation methods set local threshold while computing curvature. It is difficult to establish a single global threshold[3–6].

Major research work has been carried out in extracting volumetric and free-form features in the last two decades.However, most feature recognition (FR) tools work on B-rep models.Innovative 3D design and manufacturing methods are mesh based[7,8].A need exists to develop FR from the mesh model.STL format is globally supported by all CAD/CAM system which makes STL a platform-independent data exchange[9].If we recognize features from STL model, it will be a unique data translator utility [10,11].

The above observations inspire the research work reported in this paper.We focus on implementing elegant hybrid mesh segmentation algorithm for compound hole recognition from CMM. This paper adopts a hybrid mesh segmentationapproach for detecting intersectingholes.The algorithm segments the CMM into basic primitives like plane, cylinder, cone, sphere,etc. After extraction of analytical surfaces, a rule-based approach is used for compound hole recognition. The innovation lies in the intersectinghole detection in which no tedious curvature information and edge detection techniquehasbeen used.

The main contributions of this research can be summarized as follows:

- Complex holes lyingon multiple planer regions are detected and separated successfully.
- No curvature information has been used forhole detection.
- Automatic extraction ofthe complex intersecting holehas been donealong with computation of hole parameters.
- Feature extracted without edge detection techniques.

The rest of the paper is structured as follows: Section 2 provides a comprehensive review of relevant literature,Section 3 illustrates a proposed methodology for compound hole recognition.Section 4 defines hole feature taxonomy.Section 5 describes Hybrid mesh segmentation.Section 6 illustrates hole recognition. Discussion based on results are provided in Section 7. Section 8 provides various case studies to demonstrate intersecting hole recognition.Section 9 present conclusion and future scope

2. Literature Findings

Researchers intensively proposed and implemented various manufacturing FR system using, syntactic pattern recognition [12–14], Graph-based approach [15], volumetric decomposition based [16], hint-based [17], rule-based [18], artificial neural networks based [19,20], hybrid approach [21]. A comprehensive review of various FR approach with their strengths and weaknesses are studied in the literature [8,22–25]. Here, we limit our review of previous work upto the recognition of intersecting compound holes.

We group the related works into four categories: *primitive fitting based FR*;*Slice-based FR*,*curvature-based FR* and *Geometry-based FR*.

Primitive fitting based FR:This approach partitions the object into basic primitives based on primitive fitting clustering [26–29]. However, existing primitive extraction techniques rarely addressed complex interacting hole from CMM. Attene. et al. [26], needs visual inspection along with a number of clusters as an input parameter to perform the segmentation. However, knowing a number of clusters before feature extraction is difficult. Fig.1(b) and Fig. 1(c) shows the limitation of this approach.

Slice-based FR: This approach partitions the object by slicing. Adhikary and Gurumoorthy [30] presented an algorithm to recognize free-form volumetric features without segmentation from CMM. They used 2D slicing to identify feature boundaries. The algorithm does not depend on mesh geometricalproperties and mesh density Fig.2(c) shows the limitation of this approach. Their algorithm depends on the choice of Minimum feature dimension (MFD) and must be known in advance before feature extraction. However, the algorithm is unable to detect and extracts parameters of interacting features for test case shown in Fig. 2(a). Muraleedharan et al. [31] presented a framework to identify interacting features in CMM. They used random cutting plane approach for feature extraction. Segmentation used for separating interacting features. However, they unable to separate interacting features for test case shown in Fig. 1(c).

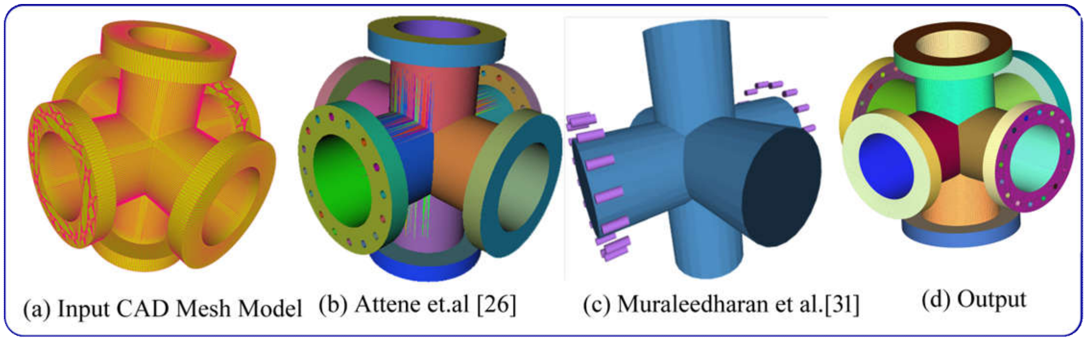


Fig.1. Failure cases for intersecting hole feature

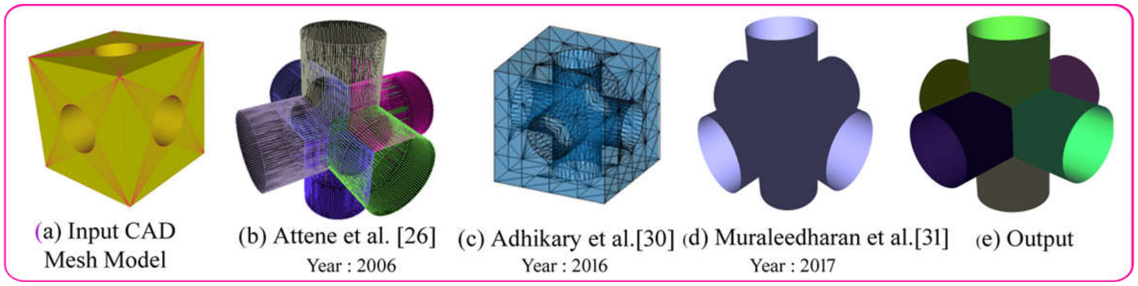


Fig.2. Failure case of Muraleedharan et al.[31] , Adhikary et al.[30] and Attene et.al [26]

Kiswanto and Azka [32] proposed a slicing technique and clustered the adjacent facets using normal vectors. The pocket and cylindrical feature automatically extracted. Adhikary and Gurumoorthy [33] presented an automatic slice-based algorithm for mesh hole filling in a CMM. However, The slice-based approach is unable to identify intersecting features [30].

Curvature based FR: STL models of mechanical parts have a diverse variety of triangles. Some researchers have taken efforts in estimating curvature along the boundary [34]. For recognizing cylinder or sphere, only curvature information (Gaussian curvature and absolute mean curvature) alone is not enough. Therefore only curvature estimates cannot be used for clustering the triangles/ facets.Computing the principal curvatures accurately is tough for coarse meshes [35]. Research shows that triangle shape has more influence on discrete curvature than triangle size does [36]. An enormous amount of time is needed for discrete curvature calculation. There are very few researchers who use facets distribution properties of the STL for FR [37].

Vorray et al. [38] proposed an algorithm to extract planar and curved surfaces using Gaussian curvature and Mean Curvature. Extracted patches were fit to B-Spline based surfaces by surface fitting techniques. However, some manual work was needed for the surface fitting of extracted patches. Lai et al. [39] presented an algorithm to detect quadratic surfaces by region growing process. However, this approach used average curvature to determine the feature type. Feature type may be wrongly computed for small regions. Xu et al. [40] proposed a method for machining FR from "In process Mesh Model" output by NC simulation system. They used curvature based region segmentation and adjacency graph-based FR. Feature extracted were plane, pockets, slot, hole, boss, spherical cap. In another approach, Xu et al. [2] presented an algorithm for FR based on region. They used the principal curvatures for region segmentation and attributed adjacency graph for feature classification. Feature extracted were holes, chamfer, pocket, boss. However, the method depends on the concave edges in the model. Xu et al. [41] presented curvature based region segmentation FR algorithm for (IPM). They create a B-rep model by processing the IPM data; then, recognize machining features from the B-rep model using the adjacency graph. Feature extracted were Holes, pockets, slots Zhang and Conclusion

Based on the quantitative relationship between mesh quality and discrete curvature [37], we proposed a hybrid mesh segmentation (facet based + vertex based + Rule-based + machine Learning based) to partition CMM using facets area, avoiding tedious curvature estimation.

Geometry-based FR: This approach partitions the object into meaningful surface primitives by utilizing geometric property like Gaussian curvature, mean curvature, Dihedral angle, etc. for identifying feature lines [29,42–45]. Kim et al. [46] proposed a framework for FR based on a tensor voting theory by extracting a feature line. However, feature line is sensitive to noise present in CMM. Most of the times, unable to detect or separate interacting features [30,31]. Yang and Shu[47] created the data bank of the dihedral angles for the core surfaces and proposed an algorithm for FR based on the dihedral angle threshold. However, the approach highly depends on a data bank. Zhang and Huang [48] presented a method for extracting boundary edges. However, the method can recognize the feature's faces, lines, and vertex only. No specific features were recognized. Jiao and Bayyana [35] presented a framework for identifying C^1 and C^2 discontinuities in STL mesh. However, the model must have sharp edges. No specific features were identified. Zhang and Li [49] presented a face clustering based region segmentation for feature extraction using "shape histograms.". The criterion used for clustering is concavity and convexity of facets. Feature extracted were a cylinder, sphere, ring, wedge only. However, complex features like blends not extracted. Qu and Stucker [50] presented a method to recognize circular holes based on detecting closed loops. However, intersected holes feature fails if the potential hole does not have a closed loop.

2.1. Conclusion

To our knowledge, extensive research has been doneon feature recognition from the B-rep model. Several researchersapply hole recognition on the B-rep model to simplify the model before meshing [51,52]. For the CMM, only geometrical information exists, no topological information. Therefore method used by researchers in FR from B-rep model, can't be utilized for CMM.

From a review of the available literature, it is found that most of the research work is limited to simple features and complex real-world features like intersectingholes are avoided. Many researchers have reconstructed the B-rep model from mesh models, but very few proceed to Feature extraction[2]. Most of the existing algorithms pose difficulties in extracting and separating intersecting features. Very little research work has been carried out on extractingcomplex interacting holes from CMM.

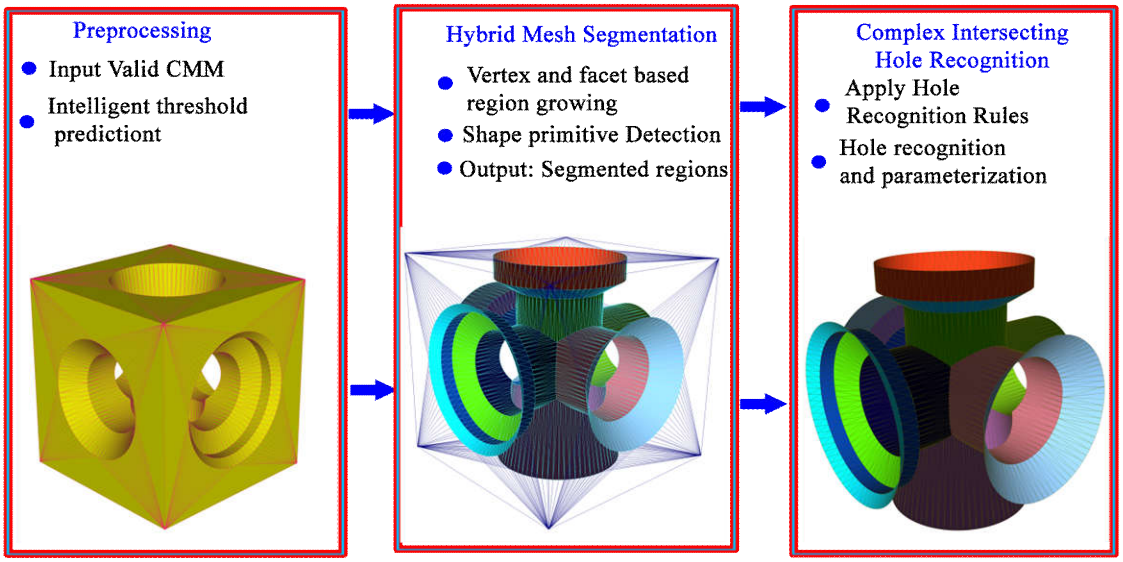
3. Methodology

The proposed algorithm involves three steps viz. preprocessing, hybrid mesh segmentation and compound hole recognition. Fig. 3 illustrates the overall strategy to extract complex intersecting holes from CMM including hole chain interaction which consists of following steps:

3.1. Preprocessing

3.1.1. Input CAD Mesh Model

The proposed method takes a valid CMM as input in ASCII or Binary format. In this research work, an input is valid STL model free from errors, hence nomodel healing is required[11].



(a) Preprocessing (b) Hybrid Mesh Segmentation (c) Complex Intersecting Hole Recognition

Fig.3. The framework of the proposed methodology

3.1.2. Build Topology

The topology construction is crucial for feature extraction. The objective of topology construction is to create and depict the faceted data in an appropriate data structure. In preprocessing, we build topology in inputted CAD mesh model.

3.1.3. Automatic Threshold Prediction

We have proposed and implemented intelligent prediction of threshold using a machine learning classifier to partition CMM. We haveperformed supervised machine learning for the prediction of Area Deviation Factor (Threshold) automatically. A detailed description of automating threshold prediction using K-Nearest Neighbor (KNN) classifier is beyond the scope of this paper.

3.2. Hybrid mesh segmentation

Hybrid mesh segmentation (HMS)uses region growing algorithms to clusters facets into groups.The approach is hybrid as we use the facet area property to group facets together, using a combination of vertex-based and facet-based region growing algorithms[53]. After segmentation, each facet groupis subjected to several conformal tests, to identify the type of analytical surface it might be representing such as a cylinder, cone, sphere,etc

3.3.Compound holes recognition

We detect complex intersecting holes and hole chains by applying a set of rules based on adjacency information of the primitives detected in the previous step.

The various functional modules of this algorithm are discussed in details in the following sections.

4. Hole Taxonomy

4.1. Basic terminology

CAD Mesh Model: A model created by triangulating B-rep model into STL format in the CAD system.

Body: A single solid block.

*Shell:*A connected set of faces, see Fig.4.

Region: A set of connected facets having specific geometrical attributes[40].

Loop: A connected series of coedges which is an intersection border of two adjacent regions[54].

*Hole chain:*A sequence of connected holes.

Hole chain interactions: A hole belonging to one chain opens into a hole belonging to another chain[55].

4.2. The basic concept of compound hole

Compound holes are one of the most common features in a model as they are relatively easy to create/manufacture and serve a variety of purposes. Compound holes may be created as an extruded cut feature or as a revolve feature.

4.3. Compound hole taxonomy

A compound hole begins and ends with a planer or curved face. All subsequent faces of a hole share a common axis. All faces of holes are sequentially adjacent. Fig.4 shows the taxonomy of compound hole features such as a simplehole, countersunk, counterbore, counter drill, taper hole, hole chain, etc consider in this research. A holeis terminated with a valid hole bottom. Following hole end type have been recognized from CMM, see Fig. 5.

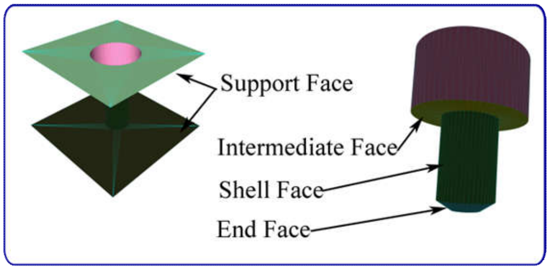


Fig.4. Taxonomy of compound hole

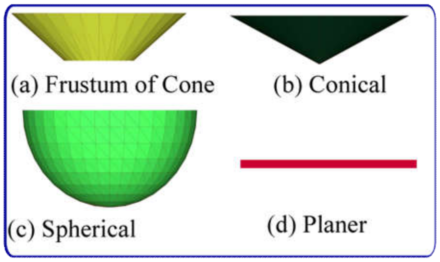


Fig.5. Hole end type

4.4. Intersecting Holes

Hole chains may intersect with each other.(see Fig. 17. (a) where 06 holes intersect with other). The present algorithm is capable of identifying all such hole chains.

4.5. Hole Types

Holesare classifiedintosimplehole, countersunk, counterbore, counter drill, taper hole. Following hole typesarerecognise from CMM. Fig.6 illustrates the classification of compound holes.

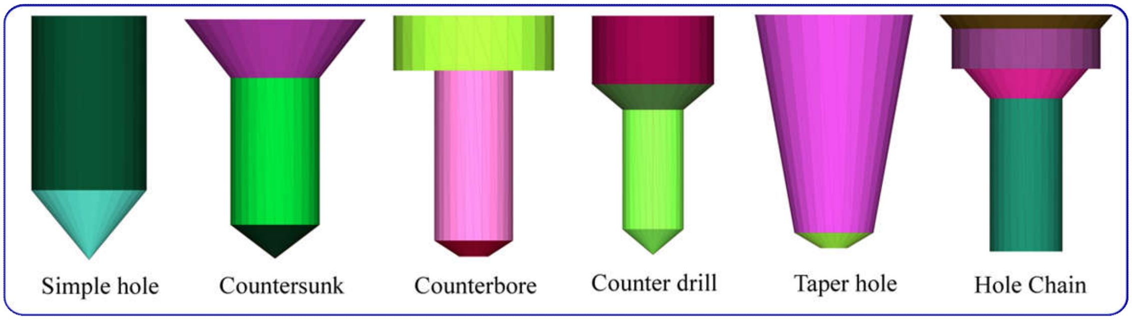


Fig.6. hole types

4.6. hole Chain

A hole chain is a set of holes that are coaxial (sharing the same axis) and connected end-to-end. The holes in a chain are ordered from the maximum radius towards minimum radius.

In this research work, the focus is on recognition of complex intersecting holes along with hole chains.

5.Hybrid mesh Segmentation

The goal of hybrid mesh segmentation is to partition CMM into basic primitives like a plane, sphere, cylinder, andcone. It is difficult to segment CMM by using facet based region growing or vertex based region growing alone. Vertex-based region growing technique is used to detect curved surface whereas Facet-based growing technique is used to detect curved features and planes. None of these approaches on their own gives a robust solution to recognize feature from CMM.

A promising approach that has become evident is a hybrid (facet and vertex based) one wherein the advantages of the above approaches are combined

A detailed description of Hybrid mesh segmentation is beyond the scope of this paper.The method automatically segments CMM into meaningful primitives.The specific strategy for primitive detection is outlined below.

5.1. Shape Primitives Detection

The hybrid mesh segmentation clusters the facets into different surface patches. The facets in the patches are subjected to several tests, to identify the type of surface it might be representing, such as a plane, sphere, cylinder, cone[56]. We get the geometric parameters of the segmented mesh feature. Initially, a facet of a surface patch does not have information about surface type it belongs to. After conformal testing, each facet of the plane, sphere, cylinder andcone patch is labeled as planar, sphere, cylinder and cone facet respectively,see Fig.7

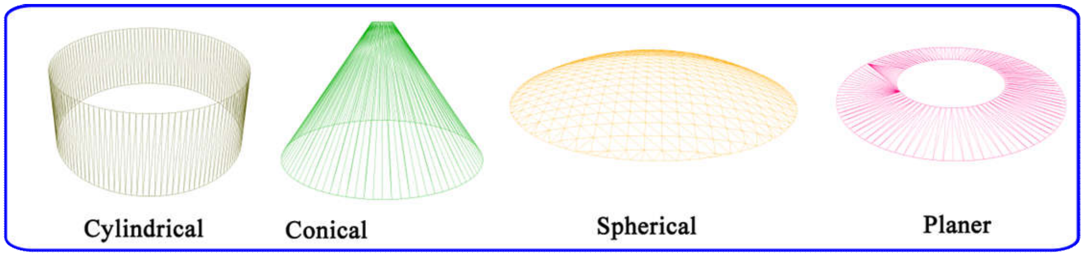


Fig.7. Shape primitives detection

5.1.1. Cylindrical surface patch Identification

From a given surface patch, the objective is to detect the set of linked triangles forming a cylindrical surface. For each point in the cylinder, the normal is perpendicular to the axis.

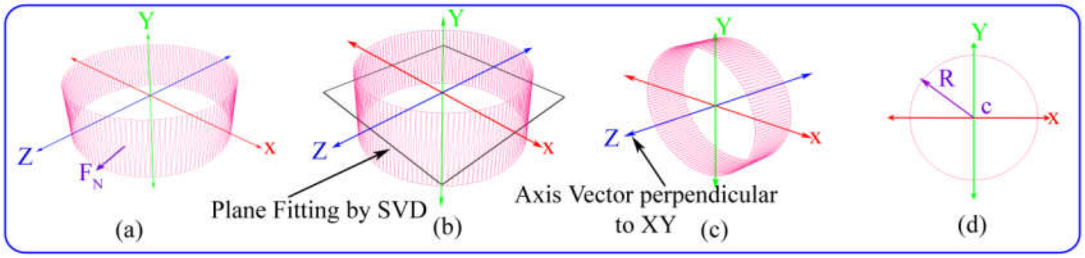


Fig.8. illustration of cylinder detection

Algorithm for cylindrical surface patch Identification

1. Calculating the axis of the cylinder :
    - i. Take the facet normals of all input triangles and fit a plane to them. The normal of this plane would be the axis, see Fig.8 (a). The method used for the fitting plane: Singular Value Decomposition (SVD), see Fig.8 (b).
    - ii. Choose eigenvector corresponding to least eigenvalue. Thiswould be the axis of the cylinder.
    - iii. After obtaining the axis, calculate rotation matrices(rotation and its inverse) that can bring the axis vector perpendicular to the XY plane,see Fig.8(c).
2. Apply the rotation matrix to the points of the input triangles. If the triangles represent a cylindrical feature, then the cylinder and its axis will be perpendicular to XY plane.
3. Take the x-y coordinates of all points and fit a circle to obtain center and radius,see Fig.8(d).
4. Store the cylinder parameters(axis, point_on_axis,radius) and create a cylinder primitive.
5. Run cylinder conformal test on all triangles to test whether they all satisfy the cylinder parameters.
6. Accept the triangles as cylinder if more than 90% triangles pass the test.

5.1.2. Cone/Frustum surface patch Identification

To get the exact normal of each point in the cone, we use the cone-apex position and the cone axis.

1. Calculating the axis cone :
    - a. Take the facet normals of all input triangles and fit a plane to them. The normal of this plane would be the axis.
    - i. The method used for the fitting plane: Singular Value Decomposition (SVD).
    - ii. There are two possible candidates for axis (i.e., Axis1 and Axis2): eigenvector corresponding to lowest and highest eigenvalue, see Fig. 9(b).
    - iii. Take dot products of both axis with the facet normals F_N, see Fig.9(c).
    - iv. Correct axis will give lots of repeating values of the dot product
        - The wrong axis will give more distinct values of the dot product.

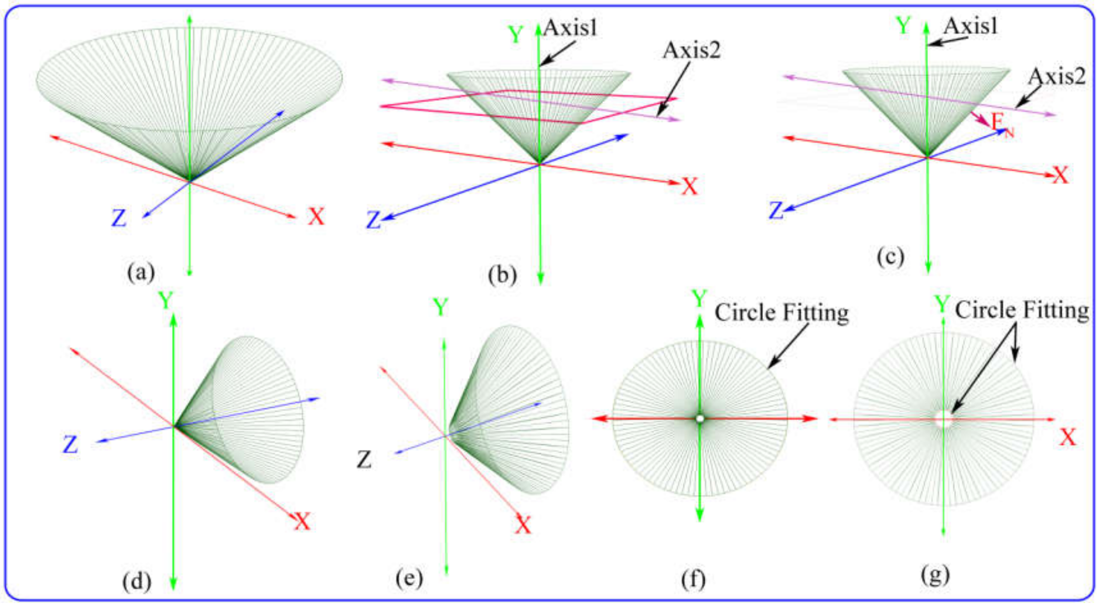


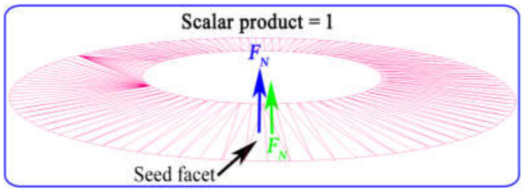
Fig.9.illustration of cone detection

2. After obtaining the axis, calculate rotation matrices(rotation and its inverse) that can bring the axis vector perpendicular to the XY plane, see Fig.9(d) for cone Fig.9(e) for frustum of a cone.
3. Apply the rotation matrix on the points of the input triangles. If the triangles represent a conical feature, then the cone and its axis will be perpendicular to the XY plane.
4. Choose the points at the extreme ends of this rotating feature.

   Fit circles at both ends and obtain center and radii, see Fig. 9(f) for a cone and see Fig. 9(g) for frustum of a cone
5. Store the cone parameters(axis, angle_of_cone, apex_point) and create a cone primitive.
6. Run cone conformal test on all triangles to test whether they all satisfy the cone parameters.
7. Accept the triangles as cone if more than 90% triangles pass the test.

5.1.3. Planar surface patch Identification

Facets forming the plane will be an adjacent and normal vector of these facets will be in the same direction (parallel to each other).The region growing algorithm use this fact for planar surface patch identification. Fig. 10 illustrates a planar region.



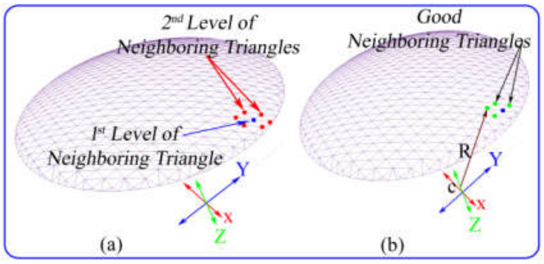


Fig.10. illustration of planar region

Fig.11. illustration of sphere detection

5.1.4. Spherical surface patch Identification

For spherical surface patch identification, select any four noncoplanar vertices. These four points determine a unique sphere. Fig.11 illustrates a spherical region.

Algorithm for Spherical surface patch Identification:

1. Iterate over the vertices in the input triangles and try to get a goodvertex such that:
    a. All first level neighbor triangles are part of the cluster
    b. All second level neighbor triangles are part of the cluster, see Fig.11(a).
2. After choosing a good vertex placed well inside the triangle cluster,
    a. Take any four neighboring vertices surrounding the vertex
    b. Calculate a circumsphere from the four points using a formula.
        i. Center and radius of the circumsphere will be obtained, see Fig.11(b).
3. Store the sphere parameters(center, radius) and create a sphere primitive.
4. Run sphere conformal test on all triangles to test whether they all satisfy the sphere parameters.
    a. if more than 90% triangles pass the test, accept
    b. If 90 % criteria are not fulfilled, go back to step 1 and try to choose another internal vertex.
    c. Set a number of retries for this(5 retries have been set).
    d. If five retries have been crossed, give up. No sphere is detected in this case.

5.2. Build Feature Adjacency

After the extraction of analytical surfaces, the next step is to build feature adjacency and topology. The adjacency information is useful in tracing adjacent features andapplying hole recognition rules. The feature adjacency is built by iterating over the boundary triangles of all detected surface patches and noting the feature on either side of the boundary

The next section deals with an algorithm to recognize complex intersecting holes for CAD mesh models, built over the segmentation workflow.

6.Compound hole recognition

The overall procedure for compound hole recognition is described below:

1. Find the set of faces that can form a hole (cylindrical and conical faces)
2. Formulate geometry and topology rules for hole recognition.
   hole forming faces are:
    a. Coaxial (sharing the same axis)
    b. Connected end to end
    c. Order of coaxial faces is such that top face has a maximum radius and bottom face has a minimum radius.
3. Identify and classify the hole feature.
4. Find hole end type (through or blind)
5. Find hole parameters

6.1. Steps in hole recognition

6.1.1. Extract cylindrical and conical faces from Face list.

hole forming faces are cylindrical and conical only. Extract cylindrical and conical typefaces from *MeshFeatures and*stores them to *HoleFaceList.*

6.1.2. Find coaxial connected Face List

Purpose: This function is used to obtained coaxial connected Face List

Input: *HoleFaceList.*

Output: *coaxial connected Face List*

Algorithm

1. Run through the *HoleFaceList*
2. Get the first*Feature* as cylindrical or conical type from *HoleFaceList.*
3. Add the selected face (cylindrical or conical) to the *coAxialFeatureList*. Get the *reference axis* and the referenceposition of the selected face.
4. From the remaining list, obtain cylindrical or conical face. Get the first and second end position of the cylindrical or conical face, see Fig.12. Get the axis.
5. Check, whether the first and second end position of cylindrical or conical face lies on *reference axis* at the same time, dot product of axis and *reference axis* as equal to 1 or –1. If these conditions satisfied by selected cylindrical or conical faces, add it to *coAxialConnectedFeatureList*.
6. Output will be to*coAxialConnectedFeatureList.*

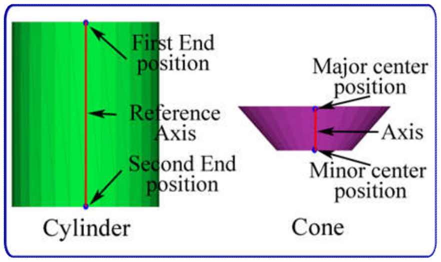


Fig.12. co-axial connected face list

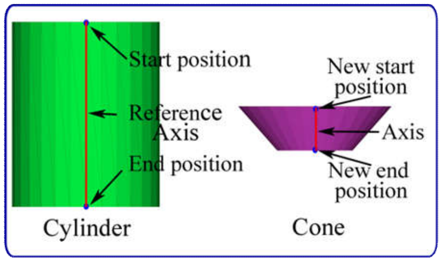


Fig.13. order co-axial connected face list

6.1.3. Find Order coaxial connected Face List

Purpose: This function is used to get *Order coaxial connected FaceList*with maximum radius is at the top and minimum Radius at bottom face. A coaxial list may have many hole features. So the hole features are separated out on the basis of connectivity. Ordering the features in a hole chain, so that correct order of faces in holes is established.

Input: *coAxialConnectedFeatureList.*

Output: *Order coaxial connected Face List*

Algorithm

1. Run through the *CoAxialFacelist*
2. Select the first face. Get the *start position* and *end position* of same,see Fig.13.
3. From the remaining list, get the other face. Get the *next start position* and *next end position*, next *start radius* and *next end radius* of same.
4. Check the *start position* and *next start position*. If both are same, insert first face to order*OrderCoAxialConnectedFaceList*. Remove the first face from the coaxial face list. update start position and start Radius i.e.

   Start position = next End position

   Start Radius = next End Radius

5. Loop repeats until *CoAxialFacelist* exhausted.
6. check *start position* and end radius
7. If *start Radius* is less than *end Radius*, reverse the order *OrderCoAxialConnectedFaceList*.
8. *Order CoAxial Connected Face list* will be prepared with maximum radius is at the top and minimum radius at the bottom.

6.1.4. Hole Recognition

Purpose: This function is used to identify and classify hole feature type from a *Order CoAxial Connected Face list.*

Input:*Order coaxial connected FaceList*

Output: *Hole Type*

6.1.4.1.Hole identification

Depending upon order of coaxially connected faces, holes are identified based on hole template matched. For example, as shown in Fig. 14(d), if Order coaxial connected *Face List* is cylinder-cone-cylinder, hole type is a counter drill. The hole end type may be planer, conical or spherical. As order matches with the desired template, a hole template for the counter drill is called.

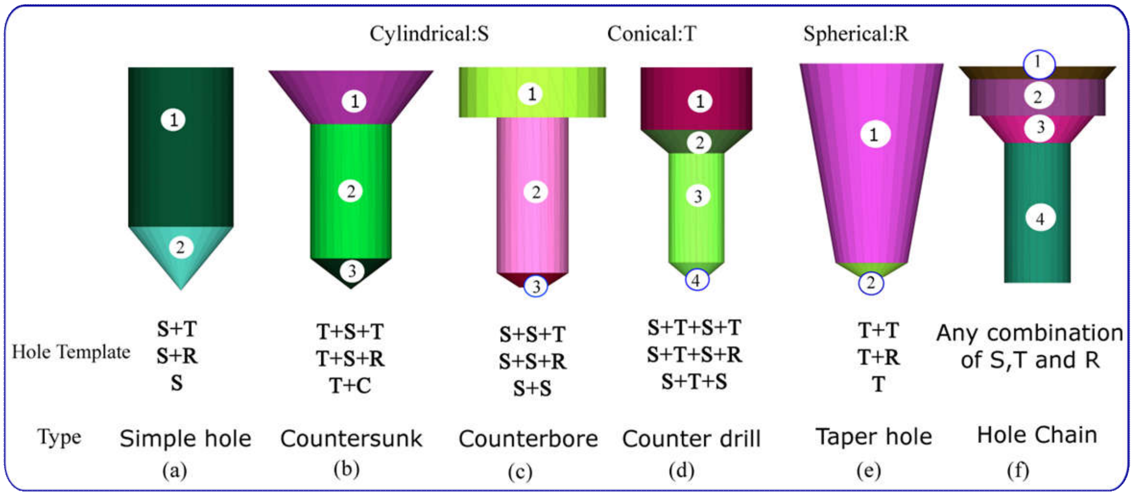


Fig.14. Hole type identification and classification

6.1.4.2. Compute hole parameters

Holetemplate computeshole parameters. For every identified hole type, the following set of parameters are calculated and stored based on hole type.

- The reference position for the hole axis, Hole diameter, hole depth, bottom angle, bottom depth, bore diameter, bore depth, sink diameter, sink depth,etc

6.1.4.3. Get the feature end type

The purpose of this function is to add hole End Type (i.e.Through or Blind) information to recognizedhole.

Algorithm

1. Run Through the order Faces.
2. Select Top Face and Bottom Face.
3. Get the internal loops for top and bottom Faces.

4. If a number of internal loops for the bottom face isone, then it is a cone or plane at hole bottom end. Hencehole end type is blind.
5. If numbers of internal loops for the bottom face are two, get *Top Face* and *Bottom Face.* If normal's of *Top Face* (F_n) and *Bottom Face*(F_n) are in the same direction, hole end type is *blind*. If normals are in opposite direction Hole End Type is *through*,see Fig.15

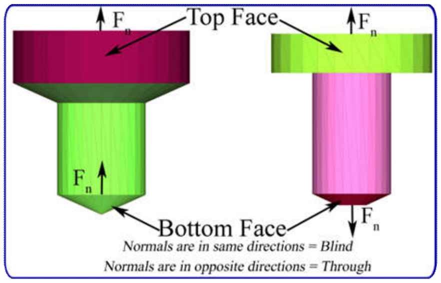


Fig.15. Feature end type

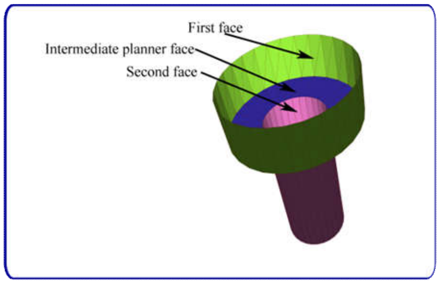


Fig.16. Add intermediate planner faces

6.1.4.4. Add the intermediate planer faces (for counterbore)

This function addsthe intermediate planner faces to the order list (for counterbore only). Get a *First face* and *second face* for a hole type counterbore.Get a face which will be common to both *First face* and the *second face* as shown in Fig 16,insert the intermediate planner face orderly,i.e., after first cylindrical face.

7.Results and discussion

Fig. 1 and Fig.2 shows failure casesexisting approach where interacting features could not get separated into individual holes, as the joints between them have a complexboundary. The proposed technique automatically extracts complex intersecting holes along with their geometric parameters.Fig.17.showsexperimental results for complex intersecting holesrecognition, tested on various benchmark test cases.

As the proposed technique is independent of feature boundary edge, results in successful detection of complexholes lies on multiple planer regions, see Fig.17(d).

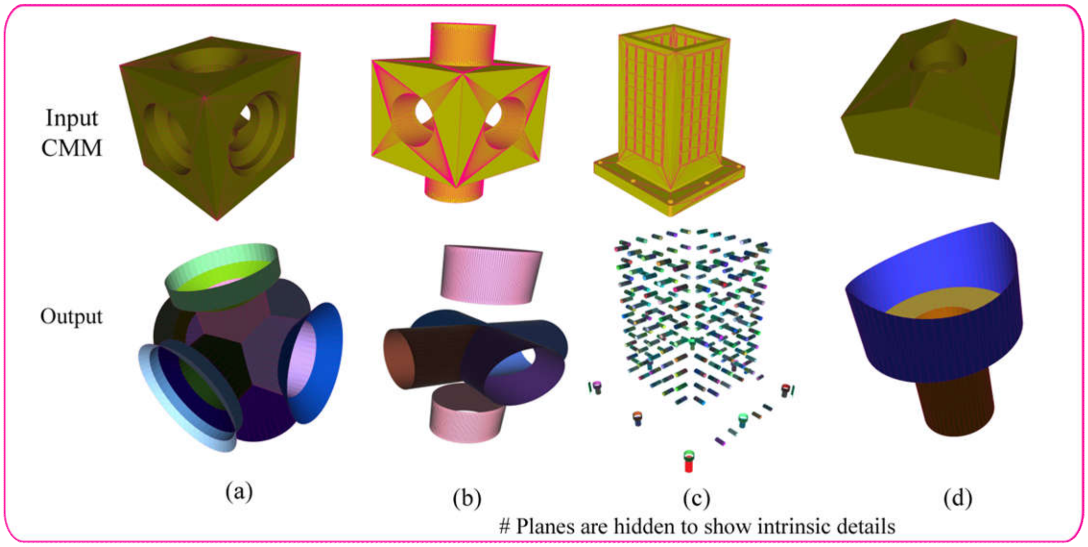


Fig.17. Experimental results for intersecting hole recognition on benchmark test cases.

Extensive experiments on CMM show that the proposed approach outperforms the existing techniques favorably and is found to be robust and consistent. The proposed technique automatically extracts intersecting holes and successfully separate the interacting holes along with their geometric parameters.

8. Implementation and Testing

The proposed algorithm has been implemented and tested on VC++ running on a computer with Intel Core i3 processor, 8GB RAM, 64bit windows 8.1 operating system. The developed system can accept any STL file generated by CAD system like AutodeskTM Inventor^{TM}2015,Solidwork TM 2017, Onshape^{TM}, etc. The system recognized following hole type as shown in Fig.18

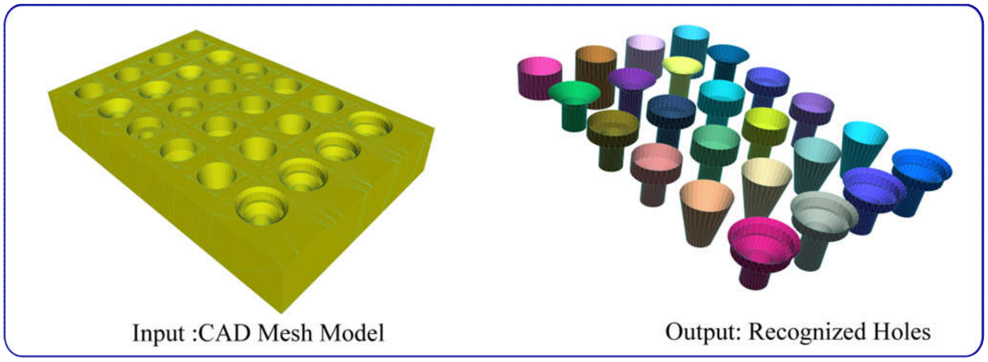


Fig.18. illustrates extracted hole types

Illustrative example

The Fig.19 illustrate interacting feature recognition for CMM. The algorithm starts with an input CMM. The STL model has 15644 facets and 7752 vertices. This part is used to test compound hole features.

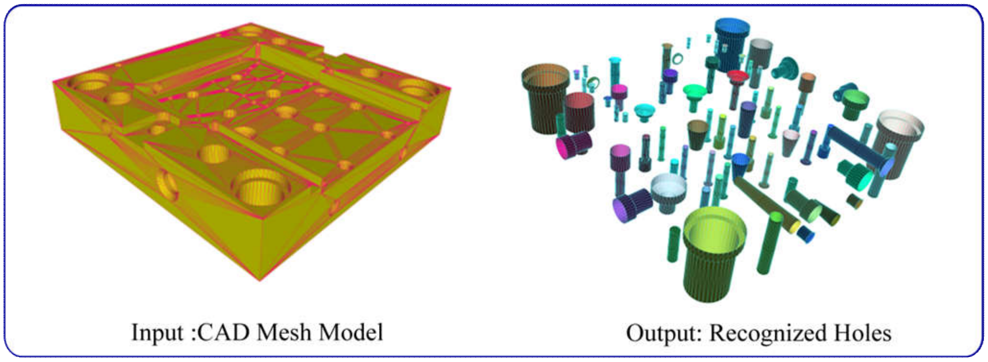


Fig.19. illustrates extracted hole feature

The system correctly extracts all regions and classifies them into compound hole features. The system takes approximately 1.767seconds for compound hole recognition. TABLE 1 illustrates extracted compound hole features with hole types.

Table 1 Case study 1: Extracted Hole types

Compound hole Type	No of instances(Blind)	No of instances(Through)
Simple hole	12	4
Counter Bore	2	16
Counter Sunk	10	14
Counter Drill	7	1
Tapered	4	2
Hole chain	3	1
Total No. of holes Recognised	**38**	**38**

9. Conclusion

In this research, an elegant method has been proposed and implemented for compound hole recognition from CMM using a hybrid region growing approach. A rule-based approach is usedfor hole recognition. The proposed algorithm captures and separates intersecting. Comparing with existing recent approaches such as Muraleedharan et al. [31], Adhikary et al.[30] on benchmark test cases, the proposed technique successfully recognized intersecting compound holes and their geometric parameters.The proposed approach is simple,more general and more reliable.

The proposed algorithm has been tested with CAD models taken from NIST repository [57] and found to be consistent in recognizing compound holes such as Simple hole, Counterbore, Countersunk, Counter drill, Tapered and hole chains along with intersecting holes.

As for our feature work, we need to develop an algorithm to detect hole interaction attributes and suppress intersecting holes.

10. Acknowledgments

This work has been supported by *Centre for Computational Technologies (CCTech), Pune, India*. We would like to thank the anonymous reviewers for their valuable recommendations.

References

[1] D. Yan, W. Wang, Y. Liu, and Z. Yang, *Variational mesh segmentation via quadric surface fitting, Computer-Aided Design, 44(11)*, 2012, 1072-1082.

[2] S. Xú, N. Anwer, C. Mehdi-Souzani, R. Harik, and L. Qiao, *STEP-NC based reverse engineering of in-process model of NC simulation, The International Journal of Advanced Manufacturing Technology,86(9-12)*, 2016, 3267-3288.

[3] T. Várady, M. A. Facello, and Z. Terék, *Automatic extraction of surface structures in digital shape reconstruction, Computer-Aided Design, 39(5)*, 2007, 379-388.

[4] P. Benkő and T. Várady, *Segmentation methods for smooth point regions of conventional engineering objects, Computer-Aided Design, 36(6)*, 2004, 511-523.

[5] P. Csákány and A. M. Wallace, *Computation of Local Differential Parameters on Irregular Meshes*, (R. Cipolla and R. Martin : Springer London, 2000).

[6] J. Huang and C. H. Menq, *Automatic data segmentation for geometric feature extraction from unorganized 3-D coordinate points, IEEE Transactions on Robotics and Automation, 17(3),* 2001, 268-279.

[7] D.Tang, L. Zheng, and Z. Li, *An Intelligent Feature-Based Design for Stamping System, The International Journal of Advanced Manufacturing Technology, 18(3)*, 2001, 193-200.

[8] J. Corney, C. Hayes, V. Sundararajan, and P. Wright, *The CAD/CAM Interface: A 25-Year Retrospective, Journal of Computing and Information Science in Engineering, 5(3)*, 2005, 188-197.

[9] M. T. Hayasi and B. Asiabanpour, *Extraction of manufacturing information from design-by-feature solid model through feature recognition, The International Journal of Advanced Manufacturing Technology,44(11-12)*, 2009, 1191-1203.

[10] F. Bianconi, *Bridging the gap between CAD and CAE using STL files, International Journal of CAD/CAM, 2(1)*, 2002, 55-67.

[11] V. B. Sunil and S. S. Pande, *Automatic recognition of features from freeform surface CAD models, Computer-Aided Design, 40(4)*, 2008, 502-517.

[12] L.K. Kyprianou, *Shape classification in computer aided design*, doctoral diss., University of Cambridge, 1980.

[13] B. K. Choi, M. M. Barash, and D. C. Anderson, *Automatic recognition of machined surfaces from a 3D solid model, Computer-Aided Design, 16(2)*,1984, 81-86.

[14] V. S. Kulkarni and S. S. Pande, *A system for automatic extraction of 3D part features using syntactic pattern recognition techniques, International Journal of Production Research, 33(6)*, 1995, 1569-1586.

[15] S. Venkataraman, M. Sohoni, and V. Kulkarni, *A graph-based framework for feature recognition, in Proceedings of the sixth ACM symposium on Solid modeling and applications*, 2001, 194-205.

[16] Y. Woo and H. Sakurai, *Recognition of maximal features by volume decomposition, Computer-Aided Design, 34(3)*, 2002, 195-207.

[17] J. H. Vandenbrande and A. A. G. Requicha, *Spatial reasoning for the automatic recognition of machinable features in solid models, IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(12)*, 1993, 1269-1285.

[18] M. R. Henderson and D. C. Anderson, *Computer recognition and extraction of form features: A CAD/CAM link, Computers in Industry, 5(4)*, 1984, 329-339.

[19] L. Ding and Y. Yue, *Novel ANN-based feature recognition incorporating design by features, Computers in Industry, 55(2)*, 2004, 197-222.

[20] R. Agarwal, V.B. Sunil , S.S. Pande, *Automatic recognition of manufacturing features using Kohonen neural network, in International Conference on Computer Aided Engineering CAE 2007, IIT Chennai*, 2007, 821-828.

[21] V. B. Sunil, R. Agarwal, and S. S. Pande, *An approach to recognize interacting features from B-Rep CAD models of prismatic machined parts using a hybrid (graph and rule based) technique, Computers in Industry, 61(7)*, 2010, 686-701.

[22] J. J. Shah, D. Anderson, Y. S. Kim, and S. Joshi, *A discourse on geometric feature recognition from CAD models, Journal of computing and information science in engineering, 1(1)*, 2001, 41-51.

[23] B. Babic, N. Nesic, and Z. Miljkovic, *A review of automated feature recognition with rule-based pattern recognition, Computers in Industry, 59(4)*, 2008, 321-337.

[24] A. K. Verma and S. Rajotia, *A review of machining feature recognition methodologies, International Journal of Computer Integrated Manufacturing, 23(4)*, 2010, 353-368.

[25] R. Zbiciak and C. Grabowik, *Feature Recognition Methods Review, in International Conference on Renewable Energy Sources-Research and Business, Proceeding, E. Rusiński,and D. Pietrusiak, Eds. Cham: Springer International Publishing,* 2017, 605-615.

[26] M. Attene, B. Falcidieno, and M. Spagnuolo, *Hierarchical mesh segmentation based on fitting primitives, Visual Computer, 22(3)*, 2006, 181-193.

[27] C. Geng et al., *A Thin-plate CAD Mesh Model Splitting Approach Based on Fitting Primitives* (EG UK: Theory and Practice of Computer Graphics, 2010), 45-50.

[28] D. Xiao, H. Lin, C. Xian, and S. Gao, *CAD mesh model segmentation by clustering, Computers & Graphics, 35(3)*, 2011, 685-691.

[29] M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, and J. Rossignac, *Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies, Proceedings of the ninth ACM symposium on Solid modeling and applications)*, 2004, 339-344.

[30] N. Adhikary and B. Gurumoorthy*, A slice based approach to recognize and extract free-form volumetric features in a CAD mesh model, Computer-Aided Design and Applications, 13(5)*, 2016, 587-599.

[31] L. P. Muraleedharan, S. S. Kannan, A. Karve, and R. Muthuganapathy, *Random cutting plane approach for identifying volumetric features in a CAD mesh model, Computers & Graphics,70*, 2018, 51-61.

[32] G. Kiswanto and M. Azka, *Automatic part primitive feature identification based on faceted models, 9(5)*, 2012, 126-132.

[33] N. Adhikary and B. Gurumoorthy, *A slice-based algorithm for automatic and feature- preserving hole-filling in a CAD mesh model, Computer-Aided Design and Applications,15(6)*, 2018, 780-795.

[34] A. Razdan and M. Bae, *A hybrid approach to feature segmentation of triangle meshes, Computer-Aided Design, 35(9)*, 2003, 783-789.

[35] X. Jiao and N. R. Bayyana, *Identification of and discontinuities for surface meshes in CAD, Computer-Aided Design, 40(2)*, 2008, 160-175.

[36] Y.Peng, C.Gao,W. He, *Research on the relationship between triangle quality and discrete curvature, China Mechanical Engineering, 19(20)*, 2008, 2459-2462, 2468.

[37] Y. Peng, Y. Chen, and B. Huang, *Region segmentation for STL triangular mesh of CAD object, International Journal of Sensor Networks, 19(1)*, 2015, 62-68.

[38] R. A. Vorray, G. O'Driscoll, and A. *Steed, Reverse Engineering Polygonal Meshes Using Discrete Differential Geometry, Computer-Aided Design and Applications, 5(1-4)*, 2008, 86-98.

[39] H. Lai, Y. Chang, and J. Lai, *Development of feature segmentation algorithms for quadratic surfaces, Advances in Engineering Software, 40(10),* 2009, 1011-1022.

[40] S. Xu, N. Anwer, and C. Mehdi-Souzani, *Machining feature recognition from in-process model of NC simulation, Computer-Aided Design and Applications,12(4)*, 2015, 383-392.

[41] S. Xu, N. Anwer, and Q. Lihong,*Feature recognition for virtual machining, Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014*, 2015, 123-127.

[42] X. Jiao and M. Heath, *Feature detection for surface meshes, Proceedings of 8th international conference on numerical grid generation in computational field simulations*, 2002, 705-714.

[43] Y. Wang, R. Liu, F. Li, and S. Endo, *An effective hole detection method for 3D models, Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, 2012, 1940-1944.

[44] S. Owen and D. White, *Mesh-based geometry:A systematic approach to constructing geometry from a finite element mesh, Proc 10th Int Meshing Roundtable, Sandia report*, 2001, 1-17.

[45] L. Chen and N. D. Georganas, *An efficient and robust algorithm for 3D mesh segmentation, Multimedia Tools and Applications, 29(2)*, 2006, 109-125.

[46] H. S. Kim, H. K. Choi, and K. H. Lee*, Feature detection of triangular meshes based on tensor voting theory, Computer-Aided Design, 41(1)*, 2009, 47-58.

[47] S. Yang and S. Shu, *Robust feature extraction for the composite surface mesh from STL file, The 9th International conference for young computer scientists*, 2008, 41(6), 1373-1378.

[48] L. Huang and X. Zhang, *Automatic identification of features from CAD models based on STL files, Applied Mechanics and Materials, 220*, 2012, 2524-2527.

[49] J. Zhang and Y. Li, *Region segmentation and shape characterisation for tessellated CAD models, International Journal of Computer Integrated Manufacturing, 29(8)*, 2016, 907-915.

[50] X. Qu and B. Stucker, *Circular hole recognition for STL based toolpath generation, Rapid Prototyping Journal, 11(3)*, 2005, 132-139.

[51] S. Venkataraman, M. Sohoni, and R. Rajadhyaksha*, Removal of blends from boundary representation models, Proceedings of the seventh ACM symposium on Solid modeling and applications*, 2002, 83-94.

[52] K. Lee, C. Armstrong, M. Price, and J. Lamont, *A small feature suppression/ unsuppression system for preparing B-rep models for analysis, Proceedings of the 2005 ACM symposium on Solid and physical modeling*, 2005, 113-124.

[53] Hase V, Verma S, Bhalerao Y, Jadhav S, Vikhe Patil, G,*Manuscripts submitted for publication*. Unpublished work.

[54] E. Abouel Nasr and A. Kamrani, *A new methodology for extracting manufacturing features from CAD system, Computers & Industrial Engineering,51(3)*, 2006, 389-415.

[55] V. Kulkarni, *Automatic recognition and suppression of complex interacting holes*, 2003.

[56] J. Wang and Z. Yu, *Surface feature based mesh segmentation, Computers & Graphics, 35(3)*, 2011, 661-667.

[57] National Design Repository, Drexel University. [Online]. Available: http://edge.cs.drexel.edu/repository/.