

# Blend Recognition from CAD Mesh Models Using Pattern Matching

Vaibhav J. Hase<sup>1, a)</sup> Yogesh J. Bhalerao<sup>2, b)</sup> Saurabh Verma<sup>3, c)</sup> G. J. Vikhe<sup>4, d)</sup>

<sup>1</sup>*SPPU, Department of Mechanical Engineering, AVCOE, Sangamner, India.*

<sup>2</sup>*Department of Mechanical Engineering, MIT Academy of Engineering, Alandi, Pune, India.*

<sup>3</sup>*Centre for Computational Technologies (CCTech), Pune, India.*

<sup>4</sup>*SPPU, Department of Mechanical Engineering, AVCOE, Sangamner, India.*

<sup>a)</sup>Corresponding author: vaibhav.hase@avcoe.org

<sup>b)</sup>ybhalerao@maepune.ac.in

<sup>c)</sup>saurabh.verma@cctech.co.in

<sup>d)</sup>drjvp@gmail.com

**Abstract.** This paper reports a unique, platform-independent approach for blend recognition from CAD mesh model using pattern matching. About 60% of the average portion of the total facets in CAD mesh model is blended features. So, it becomes essential and necessary to extract these blend features for the successful accomplishment of seamless CAD/CAM integration. The facets of the same region have similar patterns. The focus of this paper is to recognize the blends using hybrid mesh segmentation based on pattern matching. Blend recognition has been carried out in three phases viz. preprocessing, pattern matching hybrid mesh segmentation and blend feature identification. In preprocessing, the adjacency relationship is set in facets of CAD mesh model, and then Artificial Neural Networks based threshold prediction is employed for hybrid mesh segmentation. In the second phase, pattern matching hybrid mesh segmentation is used for clustering the facets into patches based on distinct geometrical properties. After segmentation, each facet group is subjected to several conformal tests to identify the type of analytical surfaces such as a cylinder, cone, sphere, or tori. In the blend feature recognition phase, the rule-based reasoning is used for blend feature extraction. The proposed method has been implemented in VC++ and extensively tested on benchmark test cases for prismatic surfaces. The proposed algorithm extracts the features with coverage of more than 95 %. The innovation lies in “Facet Area” based pattern matching hybrid mesh segmentation and blend recognition rules. The extracted feature information can be utilized for downstream applications like tool path generation, computer-aided process planning, FEA, reverse engineering, and additive manufacturing.

**Keywords.** CAD mesh model, Blend recognition, Feature recognition, Pattern matching segmentation.

## INTRODUCTION

Blends are the normal analytic or free-form surfaces which are coupled to analytical surfaces like planes, cylinders, spheres, and torus. There are many methods to create blends [1]. However, the proposed method focuses on constant radius blends. The blends are created by applying a blending operation on sharp edges. Blending operation is performed to smoothen sharp edges; improve strength; to improve safety during handling; to enhance strength by reducing stress concentration; to improve the aesthetic appearance and to ensure the manufacturability of part [2]. A typical application of blend is in Feature recognition (FR), and feature suppression. Blends often interact with each other, and with other features. Models need to be simplified for many reasons, including mesh generation, simulation, etc. [3,4].

CAD mesh models (CMM) are generated by exporting B-rep models using Computer-Aided Design (CAD) software into Stereolithography (STL) format. As CAD models exchange from the native system to the target

system, the low-level information lost. Feature recognition is a tool to recreate the feature in the target system after data exchange [5]. Majority research work carried out in extracting volumetric and free-form features in the last three decades. However, the most FR tool works on B-rep models. Innovative 3D design and manufacturing methods are mesh based [6]. A need exists to develop FR from the mesh model. STL format globally supported by all CAD/CAM system, which makes STL a medium of platform-independent data exchange [7]. Recognize features from STL model will be a unique data translator utility [8].

Literature reveals that about 60% of the average portion of the total facets in CAD mesh model is blended features [9]. So, it becomes essential and necessary to extract these blend features for the successful accomplishment of seamless CAD/CAM integration. Blends must be suppressed before performing Feature Recognition [10]. Most of the researchers use CAD models that do not have blends. Literature documents very little work on blend feature recognition from CMM. Blends are useful in downstream applications like blend suppression, which helps in recognizing volumetric features like pockets, slots, rib, etc. Blend recognition helps for cleanup operations in FEA, Mesh generation [11]. A demand necessitates developing an elegant technique for blend recognition from CAD mesh model to facilitate seamless CAD/CAM integration.

Above observations motivate the research work reported in this paper. The focus of this research paper is of blend recognition from CMM with an elegant pattern matching (PM) segmentation algorithm. The innovation lies in the mesh segmentation and recognition condition.

The main contributions of this research for extracting blend features from CAD meshes are as follows:

- Pattern matching segmentation technique dedicated to the CAD mesh.
- Intelligent combination of a vertex based, facet-based, rule-based reasoning techniques for mesh segmentation.
- Partitioning criteria used for clustering triangles is “Facet Area.”
- Feature recognition without curvature estimation.
- The algorithm does not need feature edges for feature extraction.

## LITERATURE REVIEW

In the last three decades, mesh segmentation has been comprehensively studied. The objective of mesh segmentation is to partition the input CAD mesh model into “meaningful” regions [12]. Most of the mesh segmentation methods partitioned models based on mesh attributes like Curvature, Geodesic distance, Convexity, Dihedral angle, etc. Mesh segmentation comprehensively summarized by [13–18]. However, the results of segmentation greatly depend on the choice of mesh attributes. For mechanical engineering applications, segmentation partition the CAD mesh model into a distinct, mathematically analyzable regions. Mesh segmentation is the most favored approach for extracting surface features [19].

The slice-based approach partitions the object by slicing. Adhikary and Gurumoorthy [20] presented an algorithm to recognize free-form volumetric features without segmentation from CMM. However, their algorithm depends on the choice of “Minimum feature dimension” and must be known in advance before feature extraction. Muraleedharan et al. [21] used a random cutting plane approach for feature extraction. However, their algorithm needs a number of cutting planes for FR and is assumed to be known. The feature must have the presence of inner rings. Le and Duan [18] used uniform slicing; a dimensional reduction technique which transforms 3D primitives to 2D to get a profile curve. The primitives are detected based on profile curve analysis. However, the algorithm was slice thickness dependent. Slicing techniques fail to identify or separate complex interacting features, as noted by Adhikary and Gurumoorthy [20]. Primitive fitting clustering approach partitions the object into basic primitives [22–24]. However, existing primitive extraction techniques rarely addressed blends from CMM.

Many Researchers have favored, B-rep based blend features extraction. The significant contributions are from Venkataraman et al. [3]. There have been some efforts that recognition blend features from CMM [25]. Zhang and Li [26] presents a face clustering based region segmentation algorithm using concavity and convexity of facets for automatic detection of features from the STL model. However, complex features like blends are not recognized. Garg et al. [27] presented a blend recognition and suppression algorithm of constant radius blends. However, the algorithm fails if blend faces do not have planar adjacent faces around it. Sunil and Pande [8] presented a hybrid region based segmentation framework to recognize free form features of sheet metal parts. The criteria used to identify and classify features are shape properties such as face normal, dihedral angle, Gauss, and mean curvature. However, the approach is domain-dependent and needs a user’s interaction while detecting complex parts. Gao et al.

[28] used the graph-based method for FR using a modified watershed algorithm. However, the algorithm has issues in extracting features in the model having smooth boundaries between regions.

## Literature Findings

Literature reports, least research work has been carried out on blend recognition from CMM. Most of the researchers used blend recognition and suppression from the boundary representation (B-rep) model to simplify the model before meshing. Recognizing blend features from B-rep model is easier as compared to CMM. In the B-rep model, the radius of the blend is available while radius information is not directly accessible in CMM. Most of the researchers used smooth edges as a clue for blend recognition from the B-rep model. For the STL model, no smooth edge available directly.

The curvature was most frequently used for Feature Recognition. However, for recognizing spherical and cylindrical blends, only curvature information (Gaussian curvature and absolute mean curvature) alone is not enough. For toroidal blend surfaces, curvature along profile circle is same, but curvature will be different at discrete points on the surface. Therefore only curvature estimates cannot be used for clustering the facets belonging to the blend surface.

Further, Sunil and Pande [8] have attempted to recognize hole, dimple, flange, ridge, bend while Adhikary and Gurumoorthy [20] recognized and extracted free-form volumetric features. Muraleedharan et al. [21] identified holes, slots, pockets as well as interacting features CAD mesh model, but could not recognize chamfer, fillet, and blends. From a literature review, it is evident that most of the researchers have focused on extracting limited and comparatively simple features. Recognition of complex and real-world features like blends has not been addressed so far.

## BLEND TAXONOMY

In this research work, the blend feature is considered as a region of interest from CMM. Real-world mechanical parts have blends features. The blends are created by applying blending operation on sharp edges. A blending operation on a convex edge removes material from the model is called round, whereas a blending operation on a concave edge adds material from the model is called fillet [3]. Figure 1 illustrates the blend terminology.

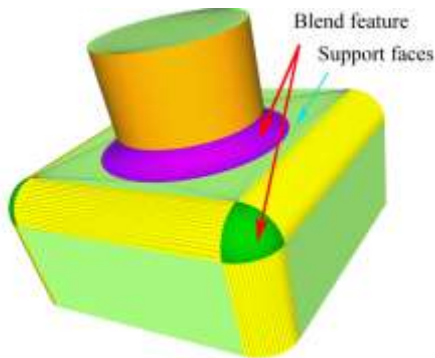


FIGURE 1. Blend terminology

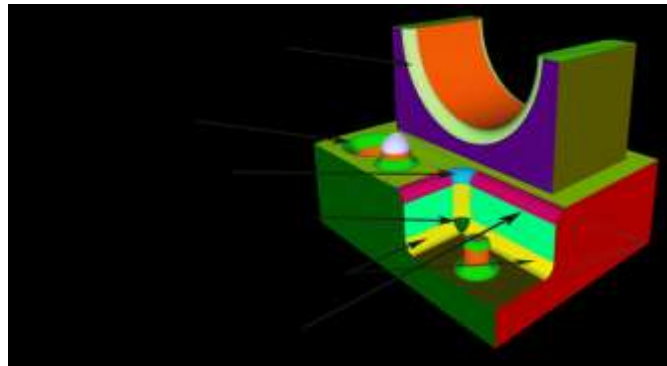


FIGURE 2. Types of Blends.

Blends are grouped into a constant radius blend, variable radius blend, and other forms. Many commercial CAD/CAM systems support both constant radius blend, variable radius blend. Constant radius blends are further classified into vertex blend, vertex toroidal blend, convex cylindrical blend, concave cylindrical blend, closed toroidal blend, and open toroidal blend. Figure 2 illustrates the types of blends.

## METHODOLOGY

The proposed algorithm involves three steps viz. preprocessing, pattern matching hybrid mesh segmentation, and blend recognition. Figure 3 illustrates the overall strategy to extract blends from CMM, which consists of the following steps:

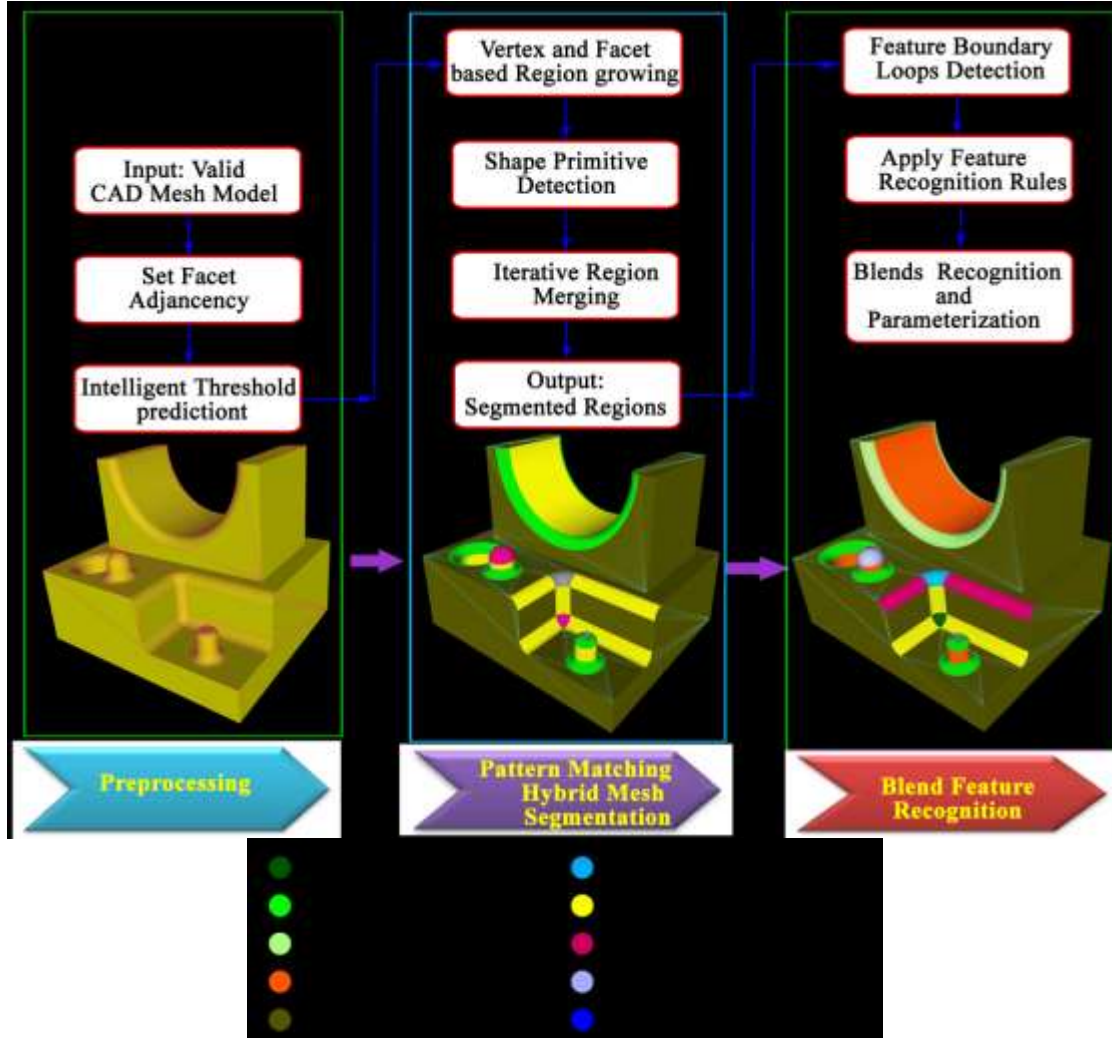


FIGURE 3. The framework of the proposed Methodology

### Preprocessing

#### *Build Facet Adjacency*

In this research work, a valid STL model which is free from errors is taken as an input. Hence there is no need of model healing [8]. In preprocessing, facets adjacency is built-in imported CAD mesh model.

#### *Intelligent Threshold Prediction*

A significant step in PM segmentation is to set the appropriate Area Deviation Factor (ADF: threshold) at the beginning. It is a cumbersome task of identifying a threshold value for getting the expected results. Most of the time, a trial and error approach is used to determine a correct threshold [21]. Inadequate threshold value leads to over-segmentation (multiple small patches) or under segmentation. Over-segmentation needs a post-processing merging

step which increases processing time whereas under segmentation leads to deficient results. However, for a layman, setting the appropriate threshold is too complicated. Manual prediction is laborious and errors prone. Therefore, an automatic and intelligent prediction approach is of significance [29].

The proposed algorithm adopts a neural network-based intelligent threshold prediction developed by Hase et al. [30]. Artificial Neural Networks (ANN) predicts the threshold by considering mesh quality of CAD mesh model as an input feature vectors. Levenberg-Marquardt back propagation (LM-BP) is used to predict the threshold. The artificial network model is designed with 2-250-1 configuration with back propagation; see Fig. 4. Extensive testing on benchmarks test cases based on LM-BP model validates ANNs prediction and improves the accuracy and stability of prediction [30]. A detailed explanation of automating threshold prediction using the neural network is beyond the scope of this paper.

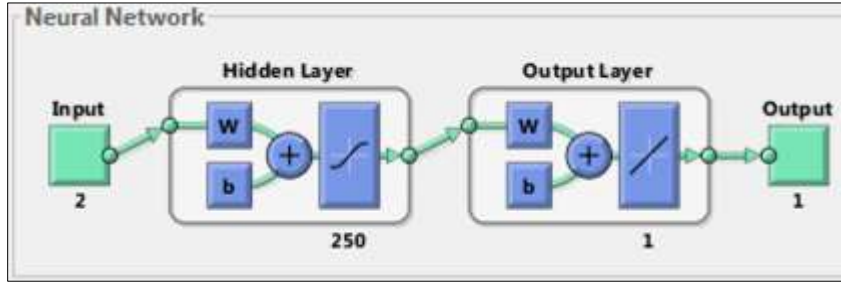


FIGURE 4. Model of the multilayer feedforward LM-BP algorithm

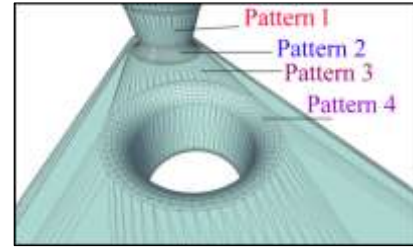


FIGURE 5. Types of the pattern.

## PATTERN MATCHING HYBRID MESH SEGMENTATION

Pattern matching (PM) is a technique of measurement of similarity between two patterns. Two patterns are similar either of magnitude or their distance between them. The proposed algorithm decomposes CAD meshes into regions. We presume that adjacent facet in a region has similar patterns. Figure 5 shows that analytical regions have different patterns. PM hybrid mesh segmentation, clusters facets in groups. To take advantage of intrinsic surface properties of the facets, the proposed method used “Facet Area” as a measurement of similarity. Figure 6 illustrates the framework of the proposed PM hybrid mesh segmentation methodology.

The objective of PM hybrid mesh segmentation is to partition CMM into basic primitives like a plane, sphere, cylinder, and tori. It is difficult to segment CMM by using facet based region growing or vertex based region growing alone [31]. None of these techniques on their own gives a robust solution to recognize feature from CMM. A promising approach wherein intelligent blending of facet-based, vertex based, rule-based reasoning are combined. Vertex-based region growing technique is used to detect curved surface [31]. The facet-based growing method is used to detect curved and planes regions not identified via vertex based region growing

Firstly, the STL model is partitioned into segmented regions that carry distinct geometrical properties. The PM hybrid approach is an intelligent blending of facet-based, vertex based, rule-based based techniques. The PM hybrid mesh segmentation uses hybrid region growing algorithms to cluster facets into groups. After segmentation, each facet group is subjected to several conformal tests to identify the type of analytical surfaces such as a cylinder, cone, sphere or tori. After extraction of analytical surfaces, feature boundaries are identified.

Region growing is a technique to group facets with the matching “Facet Area” into one region. The growing begins with a seed vertex and then a seed facet and grows from these seeds by merging neighboring facets to one region that satisfy a “Mean Area.” The procedure is iterative and reiterates until no more facets are merged.

The triangles in the clusters are subjected to several tests, to identify the type of surface it might be representing, such as a plane, sphere, cylinder, cone, and torus. For cylindrical, spherical or tori surface patch identification, Hase et al. [31] methods are used. Once feature patches are segmented, then feature parameters are determined, and feature adjacency is built.

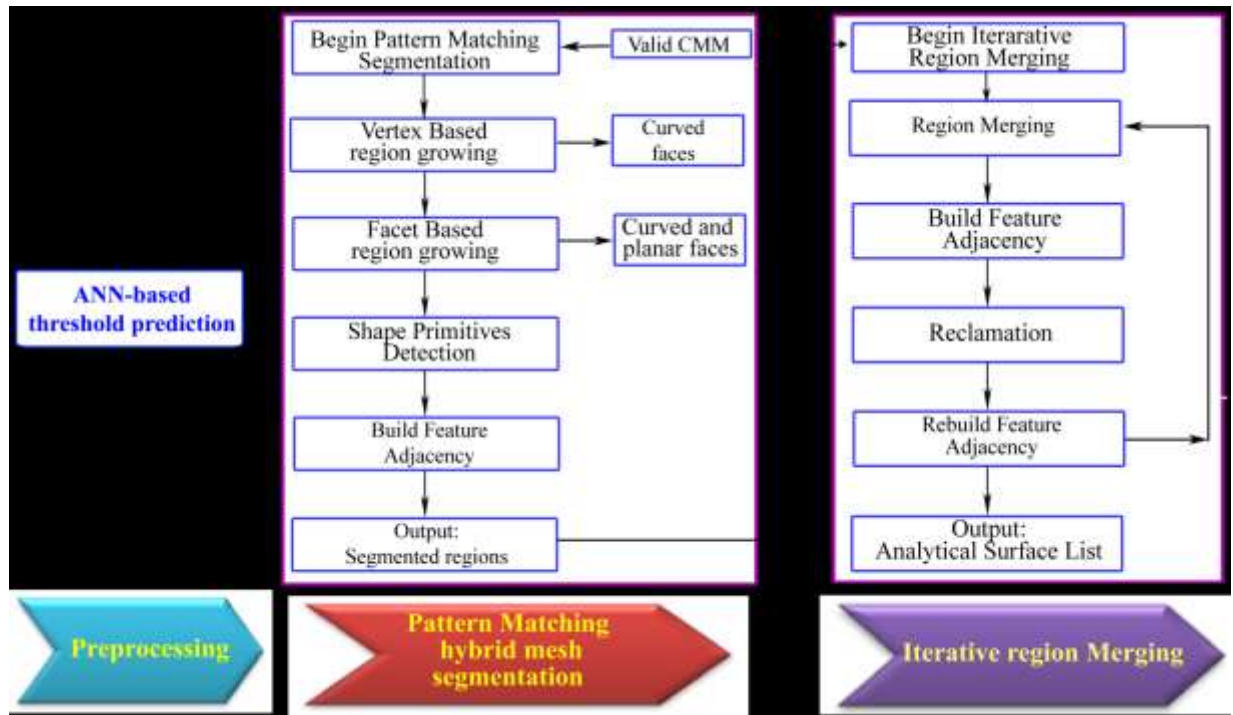


FIGURE 6. illustrates a structure of proposed PM hybrid mesh segmentation methodology

The PM Hybrid mesh segmentation leads to over-segmentation. The over segmented regions are need to be merged again to generate a single region. The proposed iterative region merging technique is based on geometry equality test. The iterative region merging technique repeatedly merged adjacent regions that have similar geometric property. After region merging, small cracks are observed close to the corner and at the region boundaries [32]. To make a watertight model, these uncollected facets are reclaimed into the surrounding identified regions (Feature) based on reclamation criteria. The developed hybrid mesh segmentation automatically segments CMM into meaningful primitives with parameters. Figure 7 illustrates examples of the cylindrical regions generated by the PM hybrid mesh segmentation. Figure 7(a) is the input mesh models; Fig. 7(b) illustrates the PM segmentation results (12 planes and 523 cylindrical patches); Fig. 7(c) shows the results of the iterative region merging process; Fig. 7(d) demonstrates the reclamation results and Fig. 7(e) illustrates the final region merging after reclamation (12 planes and 50 cylinders). Hase et al. [31] have reported a detailed description of hybrid mesh segmentation.

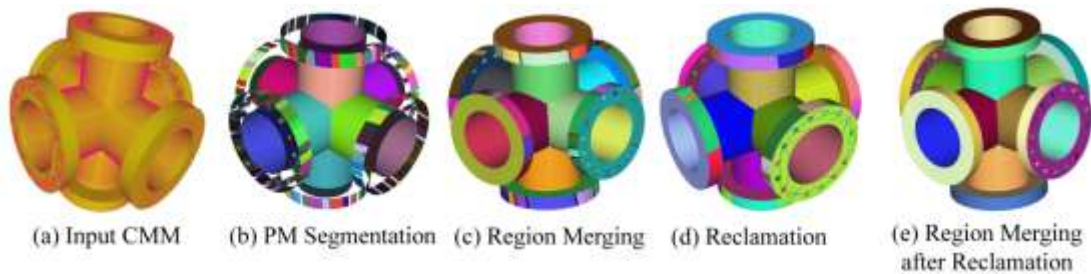


FIGURE 7. Hybrid mesh segmentation process [30]



## BLEND RECOGNITION AND PARAMETERIZATION

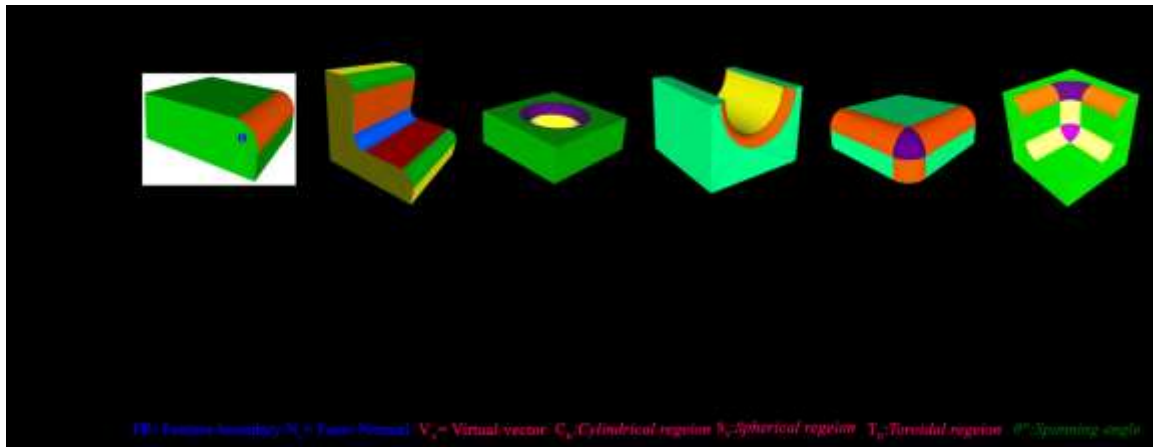


FIGURE 8. Blend type identification and classification

In this research work, the rule-based reasoning approach is used for blend feature extraction from CMM. To extract blend features, geometrical and topological rules are applied on extracted surfaces. Figure 8 shows the geometrical and topological rules for blends recognition. The feature boundary (connected series of coedges) is used to classify blend feature. Firstly, feature boundaries are detected in extracted geometry, which is intersection border of two adjacent regions. The concave or convex cylindrical blend has one feature boundary whereas, the closed toroidal blend has two feature boundaries. Later, cylinder convexity is set. Cylindrical concavity or convexity is decided by dot product principle. A virtual vector ( $V_v$ ) is a constructed between centroid and center of the cylinder. If the dot product of the surface normal vector of the cylindrical surface ( $N_F$ ) and the virtual vector is positive, the surface type is concave otherwise convex (see Fig. 9).

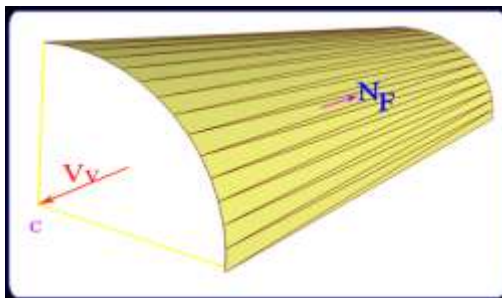


FIGURE 9. Cylinder Convexity

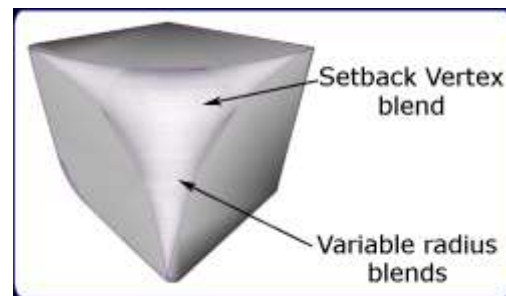


FIGURE 10. Challenges for the proposed technique

### Future Challenges

As this research work can handle only constant radius blends, future work is to recognize more complex features like variable radius blends, setback vertex along with feature interaction and suppression (see Fig. 10). Blends on nonlinear surfaces have to be addressed. As for our future work, we need to develop a deep learning approach for FR from CMM and develop a specific application which linked extracted feature information to downstream applications.

## IMPLEMENTATION AND TESTING (RESULTS)

The proposed algorithm has been executed and tested using VC++ running on a computer with Intel Core i3 processor, 8 GB RAM. The developed system can accept any STL file generated by Autodesk™ Inventor™ 2018. The efficacy of the algorithm has been tested on realistic STL CMM taken from NIST repository [33].

Fig. 11 to Fig. 13 briefly illustrates all stages in blend feature recognition for CMM. The algorithm starts with an input CMM. The STL model has 9344 facets and 5362 vertices. The test case is used to verify the efficacy of the proposed algorithm.

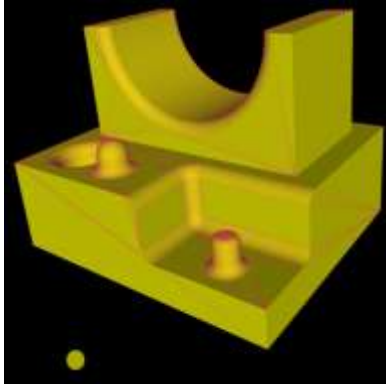


FIGURE 11. illustrates Input CMM

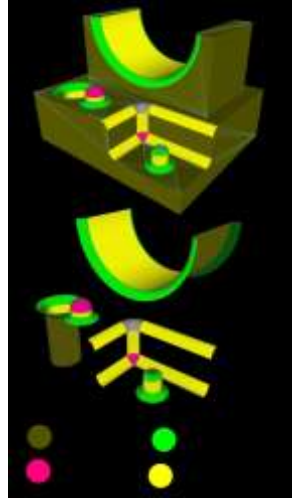


FIGURE 12. illustrates the segmented mesh model

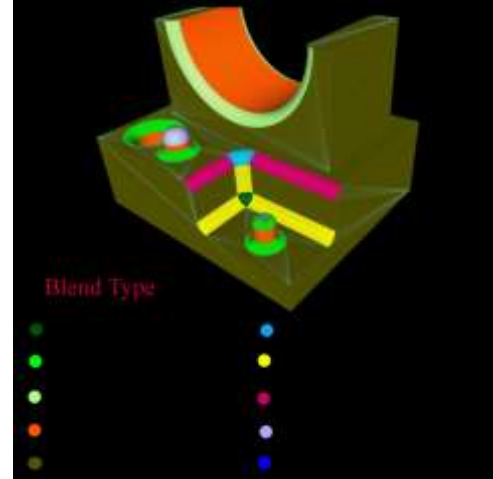


FIGURE 13. illustrates extracted blend feature.

The PM segmentation has an outcome for the test case is shown in Fig.11. The system identifies seventeen planar, nine cylindrical, two spherical, and seven torus surface. Figure 12 illustrates the segmented mesh model with each analytical surface has different colors.

After segmentation, the algorithm classifies these analytical surfaces into blend features by applying geometrical and topological rules. The system identifies one vertex blend; one vertex toroidal blend; two convex cylindrical blend; three concave cylindrical blend; four close toroidal blend and two open toroidal blends. Figure.13 illustrates extracted blend feature in different colors.

The system correctly extracts all regions and classifies them into blend features. The system takes less than one minutes for blend recognition. Table 1 illustrates segmented clusters and a number of their instances. Table 2 illustrates extracted blend features and a number of their instances.

TABLE 1. Extracted segmented clusters types

Cluster Type	Number of instances
Plane	17
Cylinder	9
Sphere	2
Torus	7
<b>Total</b>	<b>35</b>

TABLE 2. Extracted Blend types

Blend Type	Number of instances
Vertex Blend	1
Vertex Toroidal blend	1
Convex Cylindrical blend	2
Concave Cylindrical blend	3
Close Toroidal blend	4
Open Toroidal blend	2
<b>Total no. of Blends recognized</b>	<b>13</b>



To test the algorithms ability to recognize interacting blend features, test cases have been prepared. Using random color for different primitives, features can be interpreted. Figure 14 illustrates the interacting blend feature recognition.

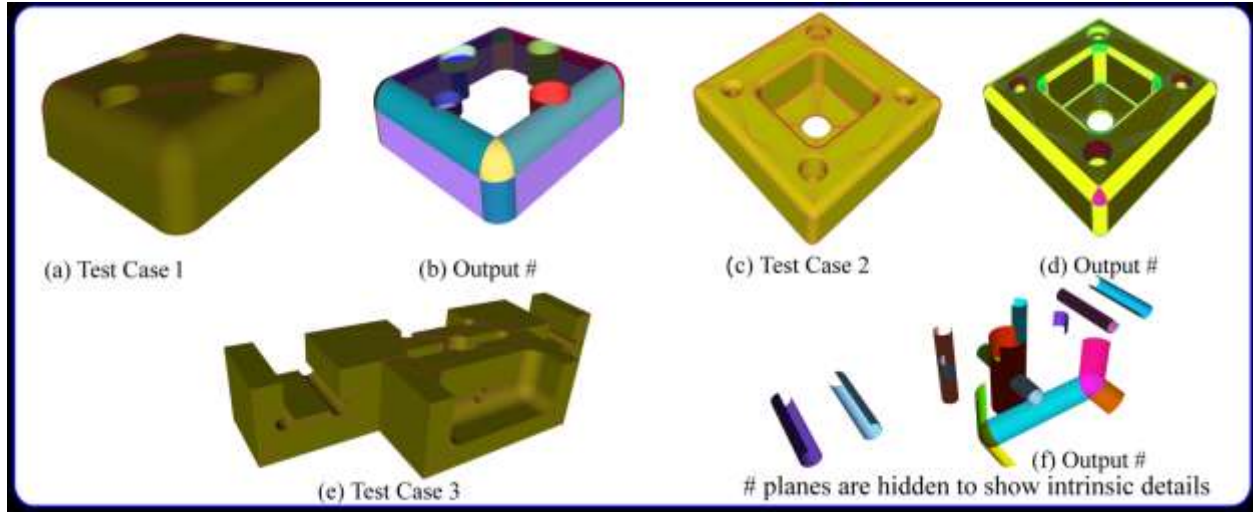


FIGURE 14. illustrates the interacting blend feature recognition

Table 3 summarizes the performance measure for a proposed algorithm for the test cases shown in Fig. 14(a), 14(c), and 14(e). The percentage coverage has been used to measure of success indicator for a hybrid mesh segmentation algorithm. It is a ratio of a number of features recognized to the number of features present in a CAD mesh model.

TABLE 3. A quantitative comparison of CAD Mesh Model

Test Cases	F	V	S	Adf	NRbrm	NRarm	T	C
Figure 14(a)	1380	690	0.349	0.75	36	25	0.254	<b>99.28</b>
Figure 14(c)	12068	6034	2.23	0.75	158	69	1.078	<b>100</b>
Figure 14(e)	1726	859	0.44	0.80	61	50	0.503	<b>99.07</b>

F: Number of Facets  
 Adf : Predicted Area deviation factor  
 NRbrm : Number of regions before region merging  
 NRarm : Number of regions after region merging  
 V: Number of Vertex  
 C : % Coverage  
 S: STL Size (in MB)  
 T: Overall Timing (in a second)

## CONCLUSION

In this paper, an elegant method has been proposed and implemented for blend recognition from CMM using PM segmentation. The method has a perspective of becoming a generic way of FR technology for CAD meshes as it partitions the STL model using facets area, avoiding tedious curvature and facet edge analysis. The proposed approach does not depend on attributes like curvature, minimum feature dimension, number of clusters, number of cutting planes, the orientation of model, and thickness of the slice to extract volumetric features. The proposed algorithm has been tested with CAD models taken from NIST repository and found to be consistent in recognizing blend features such as vertex blend, vertex toroidal blend, convex cylindrical blend, concave cylindrical blend, close toroidal blend, and open toroidal blend.

As the present work can handle only constant radius blends, future work is to recognize more complex features like variable radius blends, extrude, revolve, boss, and pockets along with feature interaction and suppression. Blends on nonlinear surfaces have to be addressed. As for our future work, we need to develop a deep learning approach for FR from CMM and develop a specific application which linked extracted feature information to downstream applications like CAPP, Tool path generation, FEA and reverse engineering.

## ACKNOWLEDGMENTS

This work has been supported by Centre for Computational Technologies (CCTech). Pune, India. Authors are also grateful to Dr. Truc Le, Dr. Ye Duan, for helping us to compute percentage coverage. Authors are thankful to Dr. P. J. Pawar, Dr. V. S. Gadakh, and Mr. Sandip Jadhav for their constructive, thoughtful suggestions that helped to improve the manuscript. We would like to thank the anonymous reviewers for their valuable recommendations.

## REFERENCES

1. J. Vida, R.R. Martin, and T. Varady, *Comput Des* **26**, 341 (1994).
2. S. Venkataraman, M. Sohoni, and G. Elber, in *Proc Sixth ACM Symp Solid Model Appl - SMA '01* (ACM Press, New York, New York, USA, 2001), pp. 99–108.
3. S. Venkataraman, M. Sohoni, and R. Rajadhyaksha, in *Proc Seventh ACM Symp Solid Model Appl - SMA '02* (ACM Press, New York, New York, USA, 2002), pp. 83–94.
4. K.Y. Lee, C.G. Armstrong, M.A. Price, and J.H. Lamont, in *Proc 2005 ACM Symp Solid Phys Model - SPM '05* (ACM Press, New York, USA, 2005), pp. 113–124.
5. M.W. Fu, S.K. Ong, W.F. Lu, I.B.H. Lee, and A.Y.C. Nee, *Comput Des* **35**, 979 (2003).
6. T. Lim, H. Medellin, C. Torres-Sanchez, J.R. Corney, J.M. Ritchie, and J.B.C. Davies, *IEEE Trans Pattern Anal Mach Intell* **27**, 851 (2005).
7. M.T. Hayasi and B. Asiabanpour, *Int J Adv Manuf Technol* **44**, 1191 (2009).
8. V.B. Sunil and S.S. Pande, *Comput Des* **40**, 502 (2008).
9. N. Rafibakhsh and M.I. Campbell, *J Comput Inf Sci Eng* **18**, 011006 (2017).
10. J.-Y. Lai, M.-H. Wang, Y.-K. Chiu, C.-H. Hsu, Y.-C. Tsai, and C.-Y. Huang, *Comput Aided Des Appl* **13**, 95 (2016).
11. X. Cui, S. Gao, and G. Zhou, *Comput Aided Des Appl* **1**, 421 (2004).
12. D. Yan, W. Wang, Y. Liu, and Z. Yang, *Comput Des* **44**, 1072 (2012).
13. M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal, in *IEEE Int Conf Shape Model Appl 2006* (IEEE, 2006), pp. 7–7.
14. A. Agathos, I. Pratikakis, S. Perantonis, N. Sapidis, and P. Azariadis, *Comput Aided Des Appl* **4**, 827 (2007).
15. A. Shamir, *Comput Graph Forum* **27**, 1539 (2008).
16. X. Chen, A. Golovinskiy, and T. Funkhouser, *ACM Trans Graph* **28**, 1 (2009).
17. P. Theologou, I. Pratikakis, and T. Theoharis, *Comput Vis Image Underst* **135**, 49 (2015).
18. T. Le and Y. Duan, *Comput Aided Geom Des* **52–53**, 231 (2017).
19. J. Wang and Z. Yu, *Comput Graph* **35**, 661 (2011).
20. N. Adhikary and B. Gurumoorthy, *Comput Aided Des Appl* **13**, 587 (2016).
21. L.P. Muraleedharan, S.S. Kannan, A. Karve, and R. Muthuganapathy, *Comput Graph* **70**, 51 (2018).
22. M. Attene, B. Falcidieno, and M. Spagnuolo, *Vis Comput* **22**, 181 (2006).
23. C. Geng, H. Suzuki, D. Yan, T. Michikawa, Y. Sato, M. Hashima, and E. Ohta, *EG UK Theory Pract Comput Graph* **45** (2010).
24. D. Xiao, H. Lin, C. Xian, and S. Gao, *Comput Graph* **35**, 685 (2011).
25. R. Sun, S. Gao, and W. Zhao, *Comput Graph* **34**, 556 (2010).
26. J. Zhang and Y. Li, *Int J Comput Integr Manuf* **29**, 907 (2016).
27. A. Garg, S. S. Krishnan, and B. Gurumoorthy, in *Int CAD Conf Exhib CAD'04*, (Thailand., 2004).
28. S. Gao, W. Zhao, H. Lin, F. Yang, and X. Chen, *CAD Comput Aided Des* **42**, 1178 (2010).
29. V.J. Hase, Y.J. Bhalerao, S. Verma, V.D. Wakchaure, and G.J. Vikhe Patil, *Int J Manag Technol Eng* **8**, 1426 (2018).
30. V.J. Hase, Y.J. Bhalerao, G.J. Vikhe Patil, and M.P. Nagarkar, in *ICCET 2019 4th Int Conf Comput Eng Technol* (2019), (Accepted for Publication).
31. V.J. Hase, Y.J. Bhalerao, S. Verma, S. Jadhav, and G.J. Vikhe Patil, *Int J Manag Technol Eng* **8**, 1102 (2018).
32. H.S. Kim, H.K. Choi, and K.H. Lee, *Comput Des* **41**, 47 (2009).
33. National Design Repository, Drexel University, available at <http://edge.cs.drexel.edu/repository/>