Enhancing Machining Feature Recognition in CAD Models through Transfer Learning on BRepNet

Konstantinos Gkrispanis[1]

[1]Chair of Computational Modeling and Simulation, Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany

E-mail(s): konstantinos.gkrispanis@tum.de

Abstract: Industrial products are initially designed using Computer-Aided Design (CAD) systems, which employ Boundary Representation (B-Rep) models for precision and efficiency. During the manufacturing process it is important to identify machining features in CAD models, as this enhances the ability to analyze, process and manufacture them. This study proposes a deep learning approach that employs transfer learning to enhance the identification and segmentation of these features. BRepNet [1], a novel neural network designed to operate directly on B-Reps, is adapted by integrating pre-trained weights and adding a modified segmentation head to refine feature identification. By freezing the backbone network and training only the segmentation head, the approach minimizes computational demands and accelerates training times. This makes deep learning more feasible for real-world applications requiring rapid deployment and high accuracy. Tested on the MFCAD dataset, the proposed architecture achieves near state-of-the-art accuracy, with 0.998 accuracy in feature identification for the version with the freezed backbone and 0.999 for the fully fine-tuned version. The study highlights the effectiveness of transfer learning in achieving high accuracy with fewer resources, demonstrating the potential for more efficient and robust CAD model analysis.

Keywords: Geometric Deep Learning, CAD learning, B-Rep, CAD segmentation, Machining features

Erschienen in Tagungsband 35. Forum Bauinformatik 2024, Hamburg, Deutschland, DOI: 10.15480/882.13539

1 Introduction

Computer-Aided Design (CAD) is an important tool in a wide variety of industries, especially in construction. Its ability to allow precise control and modification of highly intricate designs has transformed the creative processes. Almost any physical object that exists in the world was initially designed by means of CAD. Recent advancements in AI are set to revolutionize this field by automating many aspects of the design process. AI algorithms can analyze, interpret and even generate CAD models. However, there advancements bring challenges; traditional deep learning approaches for machining feature recognition are computationally expensive, often requiring extensive training time and significant computational resources.

This study leverages transfer learning to adapt existing deep learning models more efficiently. By modifying and freezing parts of the network, this approach significantly accelerates training times.

Various data formats are used to represent CAD models. The most common are: Boundary Representation (B-Rep), Point Clouds and Meshes. In this work, B-Rep will be utilized as it provides a precise and mathematically accurate representation of 3D objects.

Boundary Representation (B-Rep) is a method for representing a 3D shape by defining the limits of its volume. This approach describes a three dimensional volume through its limiting surfaces and as a result, it provides precise control over the object's geometry. It is a fundamental data format for CAD models, encoding a high level of parametric details. This data format allows for a detailed understanding of the complexities in CAD models. A solid is represented as a collection of connected surface elements, which define the boundary between interior and exterior points. Usually, a Vertex-edge-face graph is part of the B-Rep data structures. This graph offers a clear and organized way to represent the relationships between vertices, edges and faces.

A commonly used data structure in B-Rep to efficiently manage and navigate these relationships is the Winged-Edge. This data structure enhances the representation by storing the edges that connect vertices and edges that bound faces. It also provides direct access to neighboring elements, which is particularly useful for operations such as edge flipping, mesh refinement and geometry editing. The Winged-Edge structure is a powerful tool for handling complex geometric data as it supports complex topological queries and modifications. In Figure 1, Winged-Edge walk is illustrated.

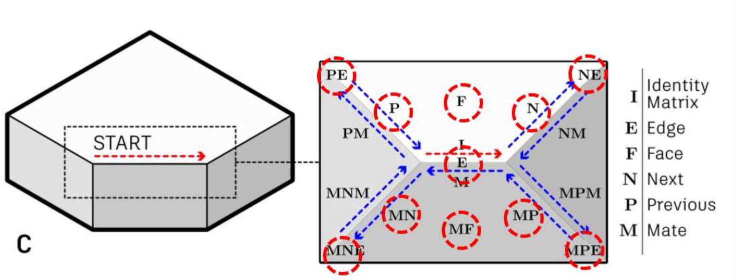


Figure 1: Winged Edge

The main contributions of this work are:

- First application of BRepNet on the MFCAD dataset, which is specifically designed for machining feature recognition.
- Significant reduction in the number of trainable parameters of the deep learning model. This reduction not only enables faster training times but also maintains high accuracy, demonstrating the effectiveness of this approach.

2 Background

Numerous models have been evaluated on the MFCAD dataset, each demonstrating notable performance. UV-NET [2], is a novel neural network architecture designed to operate directly on B-Rep data. UV-Net represents 3D CAD models by exploiting the U and V parameter domains of curves and surfaces. PointNet++ [3], is a hierarchical neural network built upon the original PointNet [4] that captures local structures in point clouds by leveraging metric space distances. Another neural network designed to learn on point clouds is DGCNN [5]. The primary goal is to leverage local geometric structure while maintaining permutation invariance, which is crucial for point cloud processing tasks such as segmentation.

Table 1: Segmentation accuracy per face and number of parameters for MFCAD dataset.

Network	Accuracy per face (%)	Number of parameters
UV-Net [2]	**99.95**	1.23M
PointNet++ [3]	91.35	1.42M
DGCNN [5]	90.99	0.53M
MeshCNN [6]	99.89	2.29M
Cao et al [7]	**99.95**	0.53M
Hierarchical CADNet (Adj) [7]	98.24	4.49M
Hierarchical CADNet (Edge) [7]	99.90	6.60M

MeshCNN [6], is a convolutional neural network designed for analyzing triangular meshes. It applies convolution operations on mesh edges, treating them as the basic building blocks, similar to pixels in an image. In [7], efficient representations of 3D CAD models were developed and used with graph neural networks for the recognition of machining features. Hierarchical CADNet [7] encodes information about the surface geometry and face topology of CAD models. A hierarchical graph is used to represent both the B-Rep face adjacency and the facets of a triangular mesh, capturing both the topology and geometry of the CAD model.

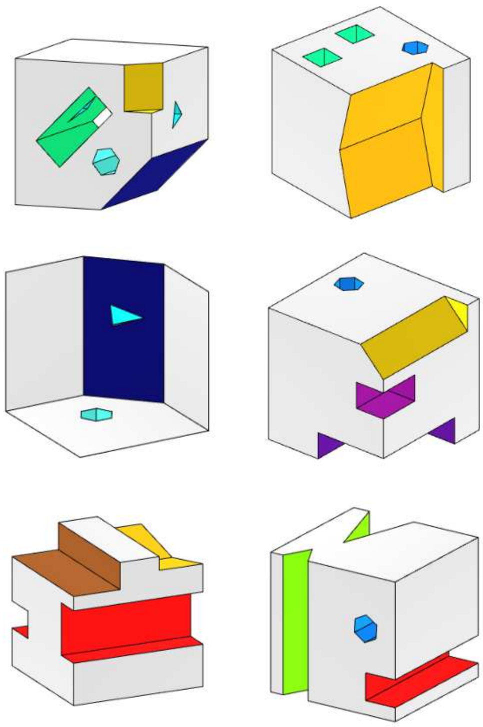
However, as illustrated in Table 1, the majority of these models possess a significantly high number of parameters, which can lead to increased computational complexity, resource consumption and longer training times. Moreover, many of these models require the conversion of the original STEP files into alternative data formats such as meshes or point clouds. This process is time consuming and also adds an additional layer of complexity to the workflow. Such transformation requirements can hinder the efficiency of the model deployment and utilization in practical applications.

3 Methodology

To tackle the challenges outlined in Section 2, a modified BRepNet architecture that significantly reduces the number of trainable parameters is introduced. This optimization is accomplished by utilizing transfer learning, where pre-trained weights from the 360Fusion dataset are loaded into BRepNet. By freezing the backbone network's weights, the training process is concentrated on an improved segmentation head, streamlining the learning process and boosting efficiency.

3.1 Dataset

MFCAD is a segmentation dataset consisting of 15,488 CAD models with labeled machining features. It was developed to enhance the automation of generating and segmenting 3D CAD models, making it a valuable tool for advancing machine learning applications in manufacturing and design [8]. In a comparative study, both graph-based and voxel-based deep learning algorithms were tested for their ability to segment CAD models. The experiments highlighted that graph-based approaches were particularly effective in identifying and recognizing complex machining features, outperforming voxel-based methods in both accuracy and efficiency. The labeled models within the MFCAD dataset provide a rich basis for further research and development in feature recognition and segmentation, with examples of these annotated models presented in Figure 2. This dataset serves as a benchmark for evaluating advanced geometric deep learning techniques applied to CAD model analysis.



Feature	Color
Chamfer	
Triangular Passage	
Rectangular Passage	
6 sides Passage	
Triangular Through slot	
Rectangular Through slot	
Rectangular Through step	
2 sides Through step	
Slanted Through step	
Triangular Pocket	
Rectangular Pocket	
6 sides Pocket	
Rectangular blind slot	
Triangular blind step	
Rectangular blind step	
Stock	

Models Classes

Figure 2: Models from MFCAD dataset and the color annotation for the classes.

3.2 BRepNet

As already discussed, the model used in this paper is the BRepNet [1]. BRepNet is another novel neural network with major contribution on this field. It is specifically designed to operate directly on B-Rep data structures. The main goal of this model is to enhance the segmentation of B-Rep models in CAD applications. BRepNet defines convolutional kernels with respect to oriented edges in the B-Rep data structures. It identifies a small collection of faces, edges and coedges in the neighborhood of each coedge, using specific learnable parameters to detect patterns in these feature vectors. The model introduces topological walks to navigate through the data structure, utilizing matrices representing next, previous and mating adjacency relationships. Kernels can have different configurations. These configurations, determine how the network captures and processes the geometric and topological information from the B-rep structure. Each kernel configuration defines a unique set of topological walks that traverse the model, aggregating information from neighboring faces, edges and coedges. From experiments in [1], it was shown that the best balance of segmentation accuracy and computational efficiency was provided by *winged edge* kernel. In [1], a dataset of CAD models with 8 machining features, named 360FusionDataset, was also presented. BRepNet's architecture is shown in Figure 3.

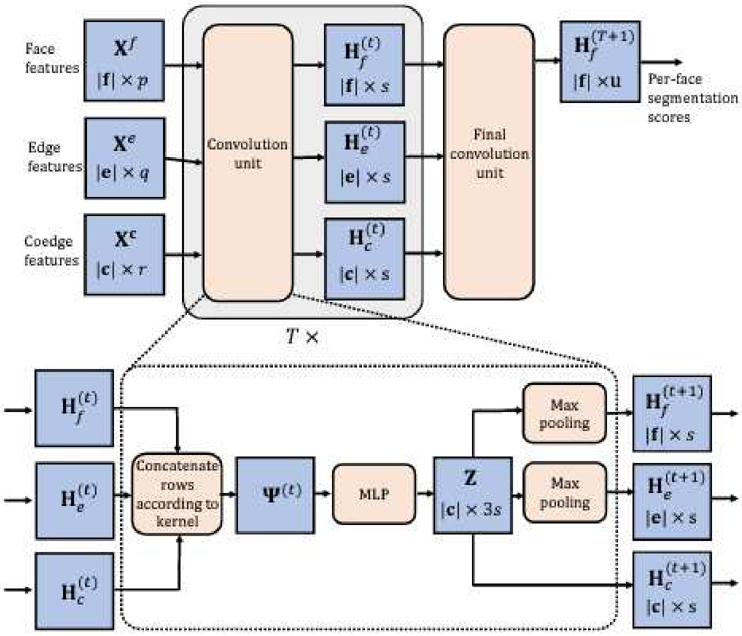


Figure 3: Original BRepNet architecture [1].

Geometric features are extracted from faces, edges and coedges, including surface types, edge geometry, edge convexity and additional flags indicating properties like edge closure and coedge direction. BRepNet achieves higher accuracy and IoU for segmentation tasks compared to methods using meshes and point clouds.

3.3 Modified Architecture & Experimental Setup

The core of the methodology lies in the modified BRepNet architecture. The original BRepNet was used as a backbone network for feature extraction, while the final convolutional unit (see Figure 3) was replaced with a custom segmentation head. This new segmentation head utilizes 1D convolutional layers, which are capable at capturing information from neighboring entities. In Figure 4, the modified architecture is plotted. To further enhance the model's performance and prevent overfitting, dropout layers were included in the segmentation head.

In order to leverage the principles of transfer learning, pre-trained weights from the 360Fusion dataset were loaded into the backbone network. These weights were frozen, ensuring that their values remained unchanged during the training phase. Only the segmentation head was subjected to training. While the original BRepNet has approximately 1.8 million parameters, the segmentation head has only approximately $142k$ parameters. By allowing only the segmentation head to be trainable, the model significantly reduces its computational requirements, enabling it to be trained on devices with limited computational resources. This technique minimizes the risk of overfitting and also speeds up the training process due to the reduced number of trainable parameters.

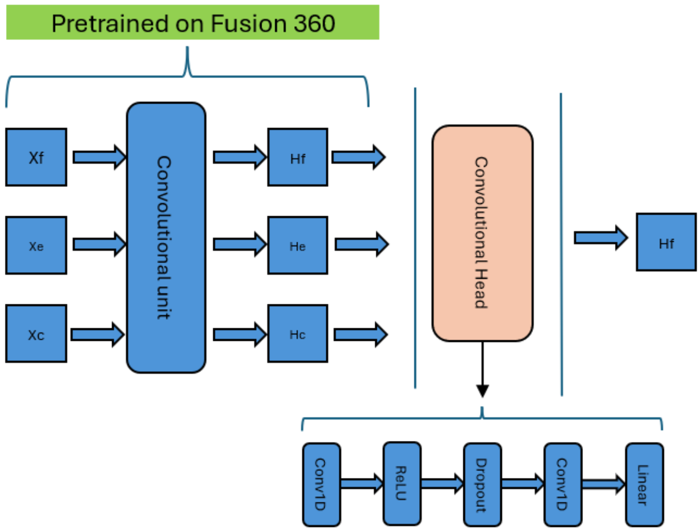


Figure 4: Modified architecture

The experiments conducted are listed below:

- Training the original BRepNet from scratch to recognize MFCAD classes.
- Fine-tuning the pre-trained BRepNet from the 360Fusion dataset on MFCAD classes.
- Using a modified architecture with pre-trained weights from BRepNet, fine-tuning only the segmentation head.

For all experiments, the learning rate was set to 0.001 and the batch size to 64. The data was split following the original paper's methodology: 60% for training, 20% for validation, and 20% for testing.

4 Results

This section presents the performance of the modified BRepNet architecture in comparison with other state-of-the-art models, as shown in Table 2.

The advantages of the proposed pipeline are clearly demonstrated. The results indicate that with significantly fewer trainable parameters, the BRepNet can achieve performance levels close to the top performing state-of-the-art models. This highlights the critical trade-off between model performance and computational efficiency.

Additionally, the benefits of transfer learning are underscored by these results. By leveraging pre-trained weights from 360Fusion dataset, the BRepNet results in faster training times and reduced computational load. The robustness and the efficiency of the proposed modification is showcased here, as this approach accelerates the training process and enables the model to generalize well.

The results in Table 3, present the performance of the three models implemented in this paper, in terms of Intersection over Union across all machining features of MFCAD dataset. It is evident that both the fine-tuned and modified versions of BRepNet achieve consistently high IoU values, often approaching or equal to 1.0. This demonstrates their robustness in accurately identifying MFCAD's classes. The

Table 2: Trainable parameters and accuracy of different models

Model	Trainable parameters	Accuracy
Finetuned BRepNet	1.8M	**0.999**
Modified BRepNet	**142k**	0.988
BRepNet from scratch	1.8M	0.975
UV-Net [2]	1.23M	0.999
PointNet++ [3]	1.42M	0.913
DGCNN [5]	0.53M	0.909
MeshCNN	2.29M	0.998
Cao et al [7]	0.53M	**0.999**
Hierarchical CADNet (Adj) [7]	4.49M	0.982
Hierarchical CADNet (Edge) [7]	6.60M	**0.999**

modified BRepNet even with fewer trainable parameters, performs comparably well to the fine-tuned version, especially in critical feature classes. The slight decreases in performance for certain classes, such as the Chamfer and Triangular through slot, will be the focus of future optimizations.

Table 3: Comparison of BRepNet models with fine-tuned and modified versions

Class	**Intersection over Union**		
	BRepNet	**BRepNet fine-tuned**	**Modified BRepNet**
Chamfer	0.995	0.995	0.986
Triangular Passage	0.998	0.999	0.998
Rectangular Passage	0.998	0.999	0.999
6 sides Passage	0.999	1.0	0.950
Triangular through slot	0.999	0.998	0.979
Rectangular through slot	0.998	0.999	0.999
Rectangular through step	0.503	0.997	0.985
2 sides through step	1.0	1.0	0.998
Slanted through step	0.521	0.997	0.988
Triangular pocket	1.0	1.0	0.998
Rectangular pocket	1.0	1.0	0.998
6 sides pocket	1.0	1.0	0.988
Rectangular blind slot	1.0	1.0	0.999
Triangular blind step	1.0	1.0	0.978
Rectangular blind step	1.0	1.0	0.999
Stock	0.999	0.999	0.999

5 Conclusion

In conclusion, this study has demonstrated that the modified BRepNet, enhanced through transfer learning, provides a robust and efficient solution for CAD model segmentation. At first, the model benefits from the feature extraction capabilities of the original BRepNet. Consequently, by freezing the backbone network and concentrating on training only a custom lightweight segmentation head, the proposed approach achieves high accuracy with reduced computational demands.

The application of transfer learning has been particularly effective, allowing for faster times and improved generalization across different CAD model datasets. It is worth mentioning, that there is no need to transform the data to other formats as the BRepNet is designed to operate directly on B-Reps.

Further works could explore optimizing the segmentation head to enhance performance in classes where IoU scores were lower than expected. Additionally, expanding the application of this model to other types of geometric data or integrating other machine learning techniques could be promising.

References

[1] J. G. Lambourne, K. D. Willis, P. K. Jayaraman, A. Sanghi, P. Meltzer, and H. Shayani, “Brepnet: A topological message passing system for solid models”, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 773–12 782.

[2] P. Jayaraman, A. Sanghi, J. Lambourne, *et al.*, “Uv-net: Learning from boundary representations”, Jun. 2021, pp. 11 698–11 707. DOI: 10.1109/CVPR46437.2021.01153.

[3] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”, *Advances in neural information processing systems*, vol. 30, 2017.

[4] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation”, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77–85. DOI: 10.1109/CVPR.2017.16.

[5] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, “Dynamic graph cnn for learning on point clouds”, *ACM Transactions on Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[6] R. Hanocka, A. Hertz, N. Fish, R. Giryes, S. Fleishman, and D. Cohen-Or, “Meshcnn: A network with an edge”, *ACM Transactions on Graphics (ToG)*, vol. 38, no. 4, pp. 1–12, 2019.

[7] A. R. Colligan, T. T. Robinson, D. C. Nolan, Y. Hua, and W. Cao, “Hierarchical cadnet: Learning from b-reps for machining feature recognition”, *Computer-Aided Design*, vol. 147, p. 103 226, 2022.

[8] *Graph Representation of 3D CAD Models for Machining Feature Recognition With Deep Learning*, vol. Volume 11A: 46th Design Automation Conference (DAC), International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Aug. 2020, V11AT11A003. DOI: 10.1115/DETC2020-22355.