# On predicting local mesh quality with machine learning

Sam Parry[1] and Steven J. Owen[2]

[1]Sandia National Laboratories*, Albuquerque, NM, USA *scparry@sandia.gov*
[2]Sandia National Laboratories, Albuquerque, NM, USA *sjowen@sandia.gov*

**Abstract**

This research note explores the use of different machine learning (ML) techniques, specifically comparing a random forest and Convolutional Neural Networks (CNN), for predicting local mesh quality in solid CAD models. Mesh quality is a critical factor in engineering simulations, and automating its prediction is a key step towards efficient Computer-Aided Design (CAD) defeaturing. The study, building on prior work, assesses the performance of these ML models, highlighting the constraints of the random forest and the adaptability of CNNs in capturing topological relationships in CAD models. The results indicate that while the random forest outperformed the CNN, the latter shows promise, suggesting a pathway for further development in ML-based CAD defeaturing processes.

## 1 Introduction.

The process of defeaturing Computer-Aided Design (CAD) models assumes a pivotal role in modern engineering simulations. Defeaturing, the act of streamlining CAD models by removing extraneous geometric intricacies, is instrumental in achieving efficient and reliable meshes for finite element analysis and other engineering simulations. Complex CAD assemblies, often laden with intricate mechanisms and details, demand meticulous user intervention and time-consuming manual model preparation to meet the prerequisites for accurate analyses.

Automating aspects of the defeaturing process with Machine Learning (ML) is an active area of research across a wide range of engineering disciplines[3, 4]. In our previous work[1], we created a system to assist users in defeaturing CAD models by suggesting defeaturing operations that will lead to the greatest improvements in mesh quality. Beginning with a solid model composed of surfaces, edges, and vertices, the system uses a random forest to predict which entities have the worst mesh quality and then provides a list of those entities to the user.

One of the constraints of the random forest is its requirement for feature information to be expressed as fixed-length vectors, limiting its ability to fully capture contextual information about an elements surroundings. Other methods like Convolutional Neural Networks (CNN) aren't bound to fixed-length feature vectors[5]. They can adapt to feature vectors of arbitrary length, allowing us to represent the topological relationships between a surface and any number of neighboring entities. In this study, we compared the performance of a new CNN implementation for mesh quality prediction against our existing random forest approach.

## 2 Methods.

We compared a random forest and CNN for predicting the local mesh quality of surfaces in a solid CAD model. Both ML methods were trained with the surface features (referred to as surface_no_op in prior work) and local mesh quality labels described in our paper on ML defeaturing[1]. The CNN was given additional information in the form of adjacency matrices that encode the topological relationships between a surface and it's surrounding edges and surfaces. Our goal was to see if a CNN adapted from BRepNet could use the adjacency matrices in conjunction with our existing feature and label set to out perform our existing random forest solution.

**2.1 Machine Learning Methods.** We implemented a CNN adapted from the BRepNet[2] architecture, which was originally designed for classifying the CAD modeling operations that created each surface. However, our CNN adaptation was reconfigured to handle regression tasks, aligning with the objective of predicting mesh quality metrics. Key differences from the original BRepNet architecture include the replacement of the final classification layer with a regression layer to suit our regression problem. Additionally, our CNN was tailored to exclusively consider surface feature vectors, eliminating the need for features from edges and

*Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

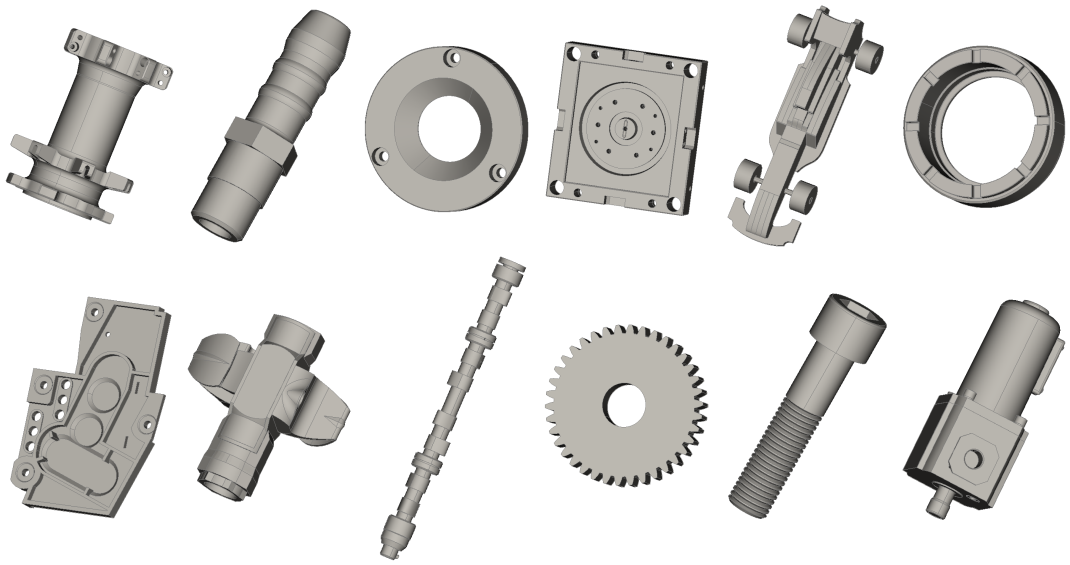

Figure 1: Some examples of the single-part CAD models contained in our 272 part dataset.

coedges, rendering it a more appropriate comparison to the random forest method, which relies solely on surface features for mesh quality prediction.

Our CNN was trained using a single input and output layer, two hidden layers, and a batch size of 20. The data split used for training, validation, and testing was maintained at a consistent 70% training, 15% validation, and 15% testing ratio, aligning with the percentages utilized in the original BRepNet architecture by Lambourne et al. This setup ensured that our CNN and RF methods were evaluated on an equal footing in the experiment. The random forest was trained with a maximum tree depth of 25, maximum forest size of 75 using a 5x2 k-fold cross validation strategy. Both methods trained until convergence used a standard L1 loss function between predictions and targets.

**2.2 Data Preparation.** We assembled a dataset of 272 CAD models from the open-source CAD library GrabCAD[7] that encompass a diverse range of real-world geometries. These CAD models represent single-volume parts commonly encountered in practice, such as gears, bolts and motors. The training data for the experiment was created in a two part process.

Part one involves using Cubit[8], our meshing and geometry toolkit, to mesh each CAD model and extract our surface features and mesh quality labels. A step-by-step guide to this process is shown below. Part two involves building the adjacency matrices that are used to represent neighboring relationships for each surface by the CNN. Building adjacency matrices involved using constructing representations of topological relationships among the surfaces and edges that make up the CAD models. These adjacency matrices will later be used by the CNN to form the convolutional kernels, a key element in the original BRepNet design. A more detailed description of this process can be found in the original BRepNet paper by Lambourne et al.[2].

1. For each CAD part in our dataset, repeat steps 2-8.
2. Import CAD part to Cubit.
3. Compute a fine target auto-size $S$ based on the part characteristics.
4. Mesh the part with tetrahedral elements of size $S$.
5. For each surface in the part, repeat steps 6-8.
6. Compute a fixed-length vector of features $X$ for the surface.
7. Compute the mesh quality metrics $M$ for the surface.
8. Write the features $X$ and labels $M$ to a new row in a .csv file containing our training data.

## 3 Results.

The random forest outperformed the CNN at predicting each of the three mesh quality metrics. While the random forest did better, the CNN was relatively successful with an R2 score above 0.85 for each metric. While theses scores aren't excellent, they do prove that the CNN approach is viable for this type of problem. It's possible that with more development, a more optimal CNN configuration would be able to close the gap in prediction accuracy.

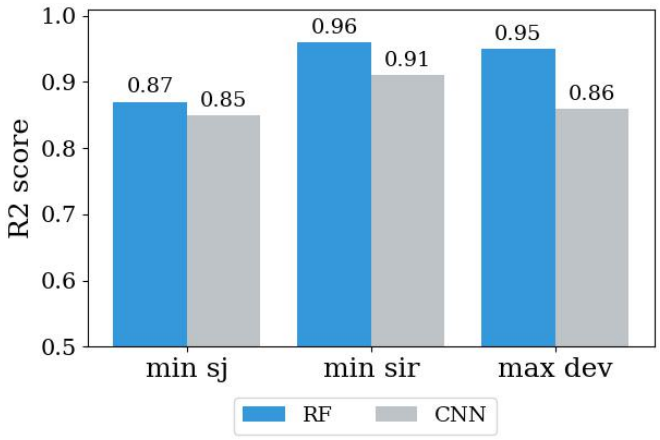


Figure 2: Model prediction accuracy's for the minimum scaled Jacobian (min_sj), minimum scaled inner radius (min_sir) and maximum deviation (max_dev) mesh quality metrics.

## References


[1] S. J. Owen, T. Shead and S. Martin, *CAD Defeaturing using Machine Learning*, in Proceedings of the 28th International Meshing Roundtable (IMR), Buffalo, NY, 2019, doi:10.5281/zenodo.3653426.

[2] J. G. Lambourne, K. D. 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, Nashville, TN, 2021, pp. 12768-12777, doi:10.1109/CVPR46437.2021.01258.

[3] F. Danglade, J. P. Pernot and P. Veron, *On the use of Machine Learning to Defeature CAD Models for Simulation*, Computer-Aided Design and Applications, 11 (2014), pp. 358-368, doi:10.1080/16864360.2013.863510.

[4] F. Danglade, P. Veron, J. P. Pernot and L. Fine, *Estimation of CAD model simplification impact on CFD analysis using machine learning techniques*, in Proceedings of Computer-Aided Design and Applications, Londres, United Kingdom, 2015, pp. 1-5, doi:10.14733/cadconfP.2015.58-63.

[5] K. O'Shea and R. Nash, *An introduction to convolutional neural networks*, preprint, arXiv:1511.08458 [cs.NE], 2015.

[6] J. Sprave and C. Drescher, *Evaluating the Quality of Finite Element Meshes with Machine Learning*, preprint, arXiv:2107.10507v1 [cs.LG], 2021.

[7] *GrabCAD*, https://grabcad.com/library

[8] *Cubit Geometry and Meshing Toolkit*, https://cubit.sandia.gov