AI-Based Retrieval to Encourage Reuse of CAD-Designs: A Methodological Study and Future Perspectives

Jakob Meinertz Grønvald[1], Morten Nørgaard[1], Negar Heidari[2], Arian Bakhtiarnia[2], Niels Henrik Mortensen[1]

[1]Technical University of Denmark
[2]Aarhus University

Abstract: In today's market, product development faces challenges such as customization demands and shortened life cycles. To address these, companies adopt product development in generations, aiming for reuse to reduce time and costs. However, escalating product portfolios lead to opacity and increased variation, hampering effective reuse. This paper proposes leveraging AI, particularly deep learning, to address unnecessary variation creation and lack of transparency. This paper investigates a deep learning methodology, UV-Net, which converts CAD solids into tensors for processing, facilitating classification and retrieval tasks. Experimental results, based on self-supervised learning with the Fusion 360 dataset, highlight AI's potential in categorizing and retrieving CAD designs. While showing promising performance, challenges persist with complex shapes and imbalanced datasets. We discuss integrating cost considerations and future research avenues for a comprehensive AI-driven design assistance system, facilitating improved decision-making in product development.

Keywords: Artificial Intelligence (AI), Computer Aided Design (CAD), Information Retrieval, Geometric Feature Analysis, Reuse

1 Introduction

In the modern market environment, companies are faced with many challenges regarding product development. Demands for customized solutions, shorter product life cycles, regulatory changes, and many other factors force companies to develop new products or product variants in a short period of time and at a low cost.

To handle these challenges many companies, turn to developing products in generations where a large part of the product can be reused in subsequent generations. As subsystems in the product can be reused, the development effort is reduced and both time and money are saved in the process (Albers et al., 2018, 2017). However, as the product offerings grow to keep up with market demands, product portfolios become increasingly less transparent, and the reuse of parts and subsystems becomes increasingly difficult. Instead of reusing parts, new variants are created, leading to unnecessary variation and increased costs (Hvam et al., 2019).

To address unnecessary variation creation the lack of transparency should be addressed. With Artificial Intelligence (AI), or more specifically through deep learning and machine learning methods, parts can be retrieved based on their geometric properties such as shapes, volume, and surface area. Such an AI system would be able to give design engineers suggestions as they create new parts for products (Krahe et al., 2022). This would reduce part variety and, more importantly, reduce development efforts and internal complexity in the company resulting in many benefits throughout the value chain (Greve and Krause, 2019).

An AI-based approach makes it possible to create a design system with some degree of automation, which otherwise would have been an immense task to fulfill due to the inconsistency and size of product databases in the mechanical engineering industry (Krahe et al., 2020b) As a result, higher costs when developing new product designs can potentially be avoided by reducing unnecessary part variant creation, e.g., by reusing existing parts and components from the part databases (Bai et al., 2010; Krahe et al., 2022; Rea et al., 2002). The time-to-market for new products can be reduced through automation of time-consuming routine tasks (Krahe et al., 2020a; Machalica and Matyjewski, 2019). Furthermore, suggesting existing solutions to the designer, which already contain product requirements, during the design process will reduce the effort of knowledge transfer between designers (Krahe et al., 2020a). Additionally, retaining the implicit experience and knowledge by reusing the existing CAD model (Computer Aided Design) ensures less risk of errors in production (Krahe et al., 2020b; Machalica and Matyjewski, 2019).

Classification of semi-finished components has also been suggested as means of optimizing the design process. By suggesting first drafts of existing similar CAD models, proposing the next design steps, and offering several alternative design options, it could enable the designer to explore the solution space more optimally. Thus, the designer is not limited to only making adaptations to existing products (Krahe et al., 2020b). Designing or customizing new products depends on the experience and creativity of engineers and designers. When different requirements between product generations and/or variants are automated, a revision of specifications and manual adaptation of CAD models to the corresponding standards will be eased (Krahe et al., 2020a). Deep Artificial Neural Network allows for the transferability of the method to new data, other product families, or different engineering domains by transfer learning or re-training. This would provide an enormous advantage in terms of generalization and scaling for efficient product design (Krahe et al., 2020a), and support designers with a design assistance system based on the design patterns from multiple classes of components and their CAD Model Tree (Krahe et al., 2021).

In this paper, we show how AI can benefit the designer in this regard by reviewing previous works focused on retrieval and classification methodologies, and by testing an existing state-of-the-art deep learning methodology, which is evaluated on a benchmark dataset, Fusion 360 dataset (Lambourne et al., 2021), Representation learning is used, converting the boundary representation (B-Rep) from the CAD files into tensors, processable by the AI. Lastly the article takes up important topics of discussion derived from the results in relation to the literature and possible additions to the methodology before elaborating on future research areas.

2 State of the Art

New developments in geometric deep learning, utilizing Boundary Representations (B-Reps), aim to enhance the level of automation in the CAD process (Heidari and Iosifidis, 2024). Various automation objectives have been explored, including reconstructing CAD models (Wu et al., 2021; Zhou et al., 2023), identifying interfaces for automatic mating in assembly models (Jones et al., 2021; Willis et al., 2022) , and transforming sketches, including hand-drawn sketches, into CAD models (Li et al., 2022; Seff et al., 2020). These automation processes leverage expert knowledge while alleviating them of repetitive and time-consuming tasks (Heidari and Iosifidis, 2024; Krahe et al., 2020b). By analyzing historical design data, geometric features can be extracted, and similarity analysis conducted across the CAD library (Heidari and Iosifidis, 2024), facilitating the identification and reduction of unnecessary or non-value-adding variations. Additionally, this approach increases the potential for reusing past designs in new products, thereby assisting designers and organizations in saving both time and money.

The design process is usually considered a highly individualized procedure with no strict definitions and is commonly based on the experience of the individual engineer (Krahe et al., 2020b). With proper retrieval and classification models it should be possible to increase reuse between product generations and across product programs (Bai et al., 2010) . Thus, a larger part of the experience from seasoned designers can be utilized by others and parts of the design process may automated inducing various benefits for the organization as mentioned earlier.

Different methods for classification and retrieval are described in the literature, and these have different variations in order to address different problems. However, four methodologies appear to recur consistently across the literature, mainly being: Point clouds, voxels, meshes, and multi-view images. These methods are elaborated in the following paragraphs:

Point clouds have been used by Krahe et al., 2022, where a cloud of points with a random distribution covers the volume or surface of a CAD model to extract geometric information to group and classify parts accordingly. However, according to Jayaraman et al., 2020 point clouds may overlook important details in the product due to their random distribution and may need to be excessively dense in order to capture them. Krahe et al., 2022, also report issues with detecting specific geometric details such as small flaps on flanges.

In a similar fashion Zhou et al., 2023 argue that common voxel grids, where a cube-shaped grid is used to approximate the object, will have lower levels of detail compared to B-Reps. Jayaraman et al., 2020 argue that voxels may be able to capture the small details of B-Reps in a CAD model by using convolutional neural networks (CNNs). However, there are limits to how high the resolution on the voxel grid can be before the cost of computing power and memory use is too high (Jayaraman et al., 2021; Kim, 2023).

Commonly, meshes are used for visualizations and, therefore, contain geometry and appearance attributes. The meshes are structured in unorganized triangles and are, therefore, commonly not watertight (enclosing the entire volume of the CAD model) (Tangelder and Veltkamp, 2008). As mentioned with the previous methods it is important to be able to capture distinct details of the product design, which means that the mesh around the CAD model should be watertight. This is possible with complex procedures or tight constraints on several factors, but many authors report problems, especially with large high-resolution meshes, in regard to memory usage, quality, and processing time (Geuzaine and Remacle, 2009; Jayaraman et al., 2021; Tangelder and Veltkamp, 2008; Vidanes et al., 2024).

Krahe et al., 2020 use a multi-view approach, where several pictures from different angles and perspectives are used to gather geometric information of the CAD model. CNNs then calculate the probability of which class the object belongs to and assign it to the one with the highest probability. Multi-view approaches have shown decent results in regard to the classification of parts. Both Krahe et al., 2019 and Su et al., 2015 reach accuracies close to 90 %, although some shortcomings are apparent, as Krahe's model wrongly categorized 18% of desks as sofas. Furthermore, Jayaraman et al., 2020 argue that because of the numerous entities in a B-Rep, multi-view images are not expressive enough and have limited application.

From the review of the literature on these four methodologies, it becomes apparent that there is a need for a method that can capture the fine details of CAD models, without becoming too costly in regard to computing and memory usage. Neither should it become overly complex with extensive add-ons or workarounds, as that will also affect performance and memory use. Jayaraman et al., 2020 have developed a method called UV-Net, which promises such characteristics. This paper tests this method on the Fusion 360 dataset (Lambourne et al., 2021) consisting of mechanical CAD designs and will be described in detail in the following section.

3 Methodology

As mentioned earlier, the input to deep learning models is in the form of structured numbers, such as vectors, matrices, and tensors. On the other hand, CAD solids are often stored in files in the B-Rep format, for instance, the STEP file format stores solids in several lines of human-readable text, where each line defines an entity such as a surface, curve, or a description of the relationship between entities (Mandelli and Berretti, 2022). Therefore, in order to process CAD solids with deep learning, they first need to be converted from B-Rep into tensors. This process is called *representation learning*.

Several approaches for CAD representation learning exist. For instance, UV-Net (Jayaraman et al., 2021) starts by creating a *face-adjacency graph* from the boundary representation, where each node represents a surface in the solid, and there is an edge between two nodes if their corresponding surfaces share a curve in their boundary. The information regarding the details of each surface and curve is stored in a 64-dimensional vector in the corresponding node or edge in the graph. To compute this information, the 3D surface is mapped into a 2D texture (image) by sampling from the surface. This map is called a *UV-grid*. Similarly, the 3D curves are mapped into a 1D UV-grid. Subsequently, these UV-grids are processed using a few convolutional layers in order to obtain the 64-dimensional vector.

Convolutional layers are fundamental building blocks in CNNs, which are a class of deep learning models commonly used for processing images. Convolutional layers utilize the convolution operation, which is a mathematical operation that combines two functions to produce a third function. In CNNs, this operation involves sliding a small window called a filter or kernel over the input data and performing element-wise multiplication between the filter and the overlapping input data, then summing up the results.

After a 64-dimensional vector for each node and edge in the face-adjacency graph is obtained, the graph is processed using graph neural networks (GNNs). Similar to convolutional neural networks that combine information in each local neighborhood of pixels, graph neural networks combine the information of adjacent nodes and edges in each layer using a message passing mechanism. After the face-adjacency graph is processed with a GNN, the resulting vectors for all nodes, called *node embeddings*, are all combined using a pooling operation in order to obtain a single vector that represents the entire graph and thus solid, called a *shape embedding*. This shape embedding can then be further processed for particular tasks such as classification, where the goal is to classify what type of object the CAD model represents, for instance, a bolt, a nail, or a screw.

We conducted a series of experiments to assess the effectiveness of UV-Net in categorizing and retrieving mechanical CAD designs. Our focus centered on training the UV-Net model in a self-supervised manner using the Fusion 360 dataset (Lambourne et al., 2021), comprised of 35,680 CAD models sourced from designs uploaded to the Autodesk Fusion 360 Online Gallery. Each CAD model in this dataset comes annotated for surface segmentation. Notably, the training process avoids annotations, relying instead on self-supervision generated through graph transformations to create positive pairs for contrastive learning.

Following UV-Net's analytical results and experimental setup, three distinct types of graph transformations are employed: Connected-patch, drop-node, drop-edge, and identity-transform. In connected-patch, a random node with its n-hop neighbors was selected as a subgraph, forming the positive pair of the graph. Drop-node and drop-edge randomly deleted nodes and edges from the graph, respectively, with a dropout probability of 0.4. In identity-transform, no graph transformation occurred, allowing the model to occasionally retain the original features of the graph itself.

UV-Net initially trained its model on a toy dataset called SolidLetters (Jayaraman et al., 2021), an annotated CAD data featuring alphabets in various fonts, for the purpose of evaluating retrieval performance using available annotations. Despite this pre-trained model being applied to Fusion 360 datasets, our attempts to replicate their results revealed a lack of generalization from the SolidLetters dataset to the Fusion 360 dataset. This discrepancy appears logical given the dissimilarity in object types between the two datasets, and the geometric features captured in SolidLetters may not adequately encompass the complexities present in a dataset comprised of mechanical objects. To address this challenge, we opted to train the model from scratch on the Fusion360 dataset and conducted retrieval tasks on the same dataset using the pre-trained model.

4 Results

Our training approach closely followed UV-Net's experimental setup. The model hyperparameters are all optimized by training the model in a supervised manner on SolidLetters dataset and evaluating the model performance. The optimized hyperparameters such as number of neural network layers, dimensions, etc. are then used for training the model on Fusion360 dataset in a self-supervised manner. Figure 1 depicts the algorithm's learning curve, showcasing loss degradation on both training and validation (test) data. While quantitative metrics like accuracy were not applicable due to the absence of annotations in the dataset, qualitative results obtained through visualization were employed to analyze performance. Subsequently, retrieval was performed on the test set of datasets. The initial step involves inputting the test CAD samples into the pre-trained model to collect their feature embeddings as the output. These feature embeddings, generated by the model's encoder, were then utilized to compute the *k* nearest neighbors of a query sample among all the test shape embeddings, using the Euclidean distance metric.

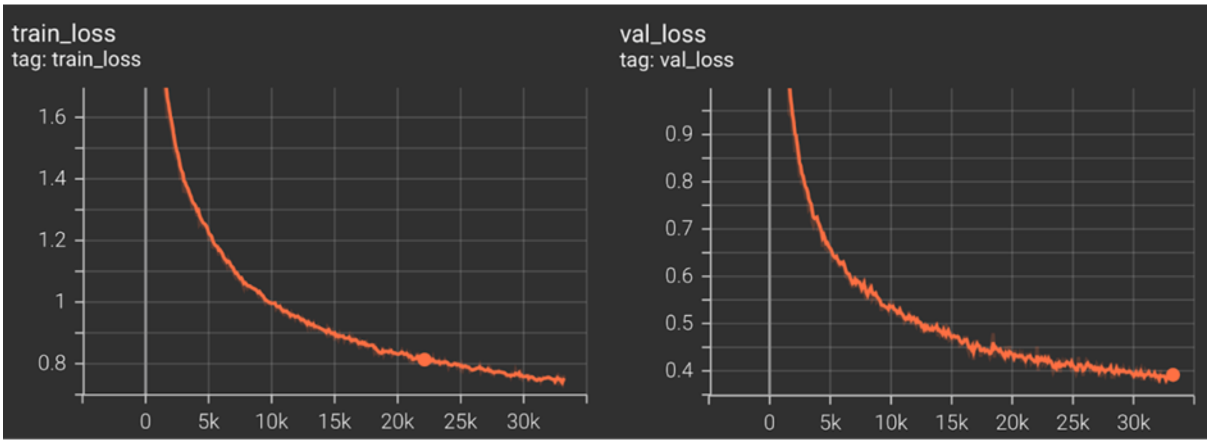

Figure 1: Loss degradation observed in both training and test data during the self-supervised training process illustrates the learning curve of the model.

Figure. 2 illustrates some example retrieval results. Notably, the model exhibits seamless performance for simple geometric shapes (rows 4, 5, 8) while encountering challenges with more complex shapes (rows 2, 9). The reason can be clarified through various factors, including an imbalance in data, lack of diversity within the dataset, or insufficiently intricate graph transformations. Although benchmark datasets like Fusion360 primarily contain geometrically simple samples, the model's capability to swiftly identify diverse design options in the initial stages of the design process can significantly save time for designers.

Figure 2: Retrieval results are obtained from the Fusion 360 Segmentation dataset, where six nearest neighbors are retrieved for each query sample.

5 Discussion

This work marks the commencement of our efforts to integrate AI algorithms into CAD design processes within industrial settings. Our next step involves generalizing this self-supervised method to datasets obtained from industrial partners. Depending on the dataset's characteristics, size, and complexity, we may need to fine-tune the model or train it from scratch. To assess the method's robustness and retrieval performance in real-world scenarios, annotating a portion of the dataset for use as a test set may be necessary. However, given the impracticality of annotating large datasets, our best practice involves adapting self-supervised methods to capture the intricate variations in CAD models used in industrial applications. As presented in the introduction, the benefits of realizing this potential could be immense when it comes to improving design engineers' product development process. Using boundary representation learning, as shown in this study, supports the above-mentioned opportunities for designers in regard to the retrieval and classification of existing parts. E.g., identifying identical simple CAD parts. Even results only on a qualitative level can support the decision-making of design engineers when it comes to reusing existing parts and retaining implicit experience-knowledge, as will be explored, and discussed in the following sections.

Although qualitative assessments may have some applicability, it leaves evaluation of the method’s accuracy more open to interpretation. However, as mentioned, the model shows decent results on simple geometries, which is supported by the improvement of the algorithm’s learning curve, but struggles with more complex parts as can be assessed in Figure 2. Generally, the results in this study leave room for improvement. If designers want to use this code to find an exact match for a specific set of specifications, be it measurements, materials, or others, they may be unsatisfied with the results. In that regard, it may not yet be the most optimal search engine to input into a PLM system, for instance. In addition, if the designer is conducting a part variance analysis, and needs to find all parts of a certain category in order to design a new standard component or conduct product program maintenance (technical updates, regulatory updates, etc.), then a high level of accuracy may not be necessary. Furthermore, if such an analysis were to be carried out on a higher level such as subassemblies or modules, the geometric similarity between systems may vary even more. In such a scenario other or additional inputs might be required to retrieve the correct samples, such as functionalities or features. However, the results of the method found in this study indicate the possibility of AI retrieval being used as a supporting tool in the design phase.

In terms of unnecessary variation creation, it could be imagined that the AI is integrated with the CAD-, PLM-, and ERP system. As the designer is creating a new part, the AI should then be able to recognize it, and based on the collection of components in the centralized database, suggest parts that already exist, which is already possible with simple geometries as shown in this paper. To give the designer the proper decision support, it would need to also present different kinds of information, such as tolerances, material specifications, measurements, and also which products the part already exists in, and its interrelation with other parts i.e. interfaces (Jones et al., 2021; Willis et al., 2022). Additionally, the cost of the part should be shown. In order to truly minimize unnecessary variation, it is not enough to only show the direct cost. A large part of overhead costs comes from increased product- or part complexity, and these should be taken into account, since direct cost will in most cases not encourage reuse (Ripperda and Krause, 2017). However, one should keep in mind that there must also be measures for when not to use the existing parts. An obvious one could be if there is no match for the specifications given. Another could be strategic considerations, though these should come from management and not the AI.

An additional challenge that may arise in real-world scenarios is imbalanced datasets, where there is a disproportionate distribution of samples in the repository. As observed in the visualized results, certain sample types may outnumber others. To address this, we propose setting a threshold, denoted as x, so that the model only retrieves neighbors with distances less than x during the retrieval process. Nevertheless, determining an optimal threshold value depends on various factors such as the distance metric and feature values. A more comprehensive solution involves balancing the dataset through data augmentation or similar techniques, a subject we plan to explore in our future work.

The result and discussion of this study brought to light several research topics that can be further developed in a new study. The self-supervised method should be tested and trained on datasets obtained from industrial partners and also document the observed benefits for designers using the trained algorithm as a step in their decision-making process. Especially the interrelation between the design engineers' decision-making and the application of machine learning algorithms as a tool to improve product solutions would be beneficial. Identifying not only the general application areas in the industry but also in small, medium, or large-scale enterprises could prove to reveal different answers.

Specifically, retrieving and classifying CAD models might help reuse of existing parts. However, it will not, at its current state, be able to assess the implications of choosing one solution over another. Optimally, a machine learning algorithm will be trained on, for example, the most cost-efficient design as suggested by Krahe et al., 2022. Yet coupling CAD models with cost data, not only considering direct cost but also including the indirect costs incurred throughout the product life cycle, is yet to be seen (Greve et al., 2022). Connecting these research areas will provide the circumstance on which design engineers can assess the most cost-optimal design and support them as a functional tool in their decision-making during product development.

6 Conclusion

The study presented in this paper was performed in order to expand upon the use of AI methods to support design engineers' decision-making in product development. The method used in this paper is called UV-net and was developed by Jayaraman et al., 2020. It was tested on a benchmark dataset without annotations. The state of the art has been explored and argues why this method may be particularly interesting to test and develop further.

The results of this study indicate the applicability of AI when it comes to supporting engineers within mechanical design. AI can be used for the retrieval of mechanical parts in CAD format, improving reuse of existing parts and avoiding spending unnecessary resources on developing existing parts and variants. Thus, reducing unnecessary product- and part-variety creation while improving product portfolio transparency.

Due to a lack of annotations in the benchmark dataset, the results are evaluated qualitatively through visualizations. The results showed that the method works well on simple part geometries but struggles with more complex shapes and systems. Factors such as, imbalanced data, lack of diversity, or insufficiently intricate graph transformation may explain this tendency, and need to be developed further before implementation is feasible. Furthermore, qualitative evaluation makes it difficult to determine accuracy precisely, but accuracy needs may vary depending on the application and require varying resources for a fully developed model. This underlines the necessity of a well-defined scope and specification of a model’s application area.

The different potential applications of an AI tool supporting the design process have been discussed in relation to the literature and the obtained results obtaining the demand both in terms of input data and accuracy. Moreover, it was found that additions to the model in the form of parameters and features, such as cost, would increase the applicability of the model for designers. In order to truly address the issues of creating product variation in the product program or to encourage the reuse of parts, indirect costs together with direct costs need to be part of the evaluation of design solutions. The discussion of this paper concludes that such additions and modifications should be investigated further in future research projects in order to truly address unnecessary variation creation and to create the most value for industry uses.

References

Albers, A., Behrendt, M., Klingler, S., Reiß, N., Bursac, N., 2017. Agile product engineering through continuous validation in PGE-Product Generation Engineering. Design Science 3. https://doi.org/10.1017/dsj.2017.5

Albers, A., Haug, F., Fahl, J., Hirschter, T., Reinemann, J., Rapp, S., 2018. Customer-Oriented Product Development: Supporting the Development of the Complete Vehicle through the Systematic Use of Engineering Generations. 4th IEEE International Symposium on Systems Engineering : October 1-3, 2018, Rome Marriott Park Hotel, Roma, Italy : 2018 symposium proceedings. https://doi.org/10.1109/SysEng.2018.8544391

Bai, J., Gao, S., Tang, W., Liu, Y., Guo, S., 2010. Design reuse oriented partial retrieval of CAD models. CAD Computer Aided Design 42, 1069–1084. https://doi.org/10.1016/j.cad.2010.07.002

Geuzaine, C., Remacle, J.F., 2009. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. Int J Numer Methods Eng 79, 1309–1331. https://doi.org/10.1002/nme.2579

Greve, E., Krause, D., 2019. Long-term effects of modular product architectures: An empirical follow-up study, in: Procedia CIRP. Elsevier B.V., pp. 731–736. https://doi.org/10.1016/j.procir.2019.04.274

Heidari, N., Iosifidis, A., 2024. Geometric Deep Learning for Computer-Aided Design: A Survey.

Hvam, L., Hansen, C.L., Forza, C., Mortensen, N.H., Haug, A., 2019. The reduction of product and process complexity based on the quantification of product complexity costs. Int J Prod Res. https://doi.org/10.1080/00207543.2019.1587188

Jayaraman, P.K., Sanghi, A., Lambourne, J.G., Willis, K.D.D., Davies, T., Shayani, H., Morris, N., 2021. UV-Net: Learning from Boundary Representations, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021. pp. 11698–11707. https://doi.org/10.1109/CVPR46437.2021.01153

Jones, B., Hildreth, D., Chen, D., Baran, I., Kim, V.G., Schulz, A., 2021. AutoMate: A Dataset and Learning Approach for Automatic Mating of CAD Assemblies. ACM Trans Graph 40. https://doi.org/10.1145/3478513.3480562

Kim, B.C., 2023. Reconstruction of design history of 3D CAD models using deep learning: research trends. JMST Advances 5, 113–119. https://doi.org/10.1007/s42791-023-00059-5

Krahe, C., Bräunche, A., Jacob, A., Stricker, N., Lanza, G., 2020a. Deep Learning for Automated Product Design. Procedia CIRP 91, 3–8. https://doi.org/10.1016/j.procir.2020.01.135

Krahe, C., Iberl, M., Jacob, A., Lanza, G., 2020b. AI-based Computer Aided Engineering for automated product design - A first approach with a Multi-View based classification. Procedia CIRP 86, 104–109. https://doi.org/10.1016/j.procir.2020.01.038

Krahe, C., Kalaidov, M., Doellken, M., Gwosch, T., Kuhnle, A., Lanza, G., Matthiesen, S., 2021. AI-Based knowledge extraction for automatic design proposals using design-related patterns. ScienceDirect. https://doi.org/10.1016/j.procir.2021.05.093

Krahe, C., Marinov, M., Schmutz, T., Hermann, Y., Bonny, M., May, M., Lanza, G., 2022. AI based geometric similarity search supporting component reuse in engineering design. Procedia CIRP 109, 275–280. https://doi.org/10.1016/j.procir.2022.05.249

Lambourne, J.G., Willis, K.D.D., Jayaraman, P.K., Sanghi, A., Meltzer, P., Shayani, H., 2021. BRepNet: A topological message passing system for solid models, in: Proceedings of the Ieee Computer Society Conference on Computer Vision and Pattern Recognition. pp. 12768–12777. https://doi.org/10.1109/CVPR46437.2021.01258

Li, C., Pan, H., Bousseau, A., Mitra, N.J., 2022. Free2CAD: Parsing Freehand Drawings into CAD Commands. ACM Trans Graph 41. https://doi.org/10.1145/3528223.3530133

Machalica, D., Matyjewski, M., 2019. CAD models clustering with machine learning. Archive of Mechanical Engineering 66, 133–152. https://doi.org/10.24425/ame.2019.128441

Mandelli, L., Berretti, S., 2022. CAD 3D Model classification by Graph Neural Networks: A new approach based on STEP format, in: Eurographics Italian Chapter Proceedings - Smart Tools and Applications in Graphics. pp. 139–149. https://doi.org/10.2312/stag.20221268

Rea, H.J., Corney, J.R., Clark, D.E.R., Pritchard, J., Breaks ML and MacLeod, R.A., 2002. Part-sourcing in a global market. CONCURRENT ENGINEERING-RESEARCH AND APPLICATIONS 10, 325–333. https://doi.org/10.1106/106329302032004

Seff, A., Ovadia, Y., Zhou, W., Adams, R.P., 2020. SketchGraphs: A Large-Scale Dataset for Modeling Relational Geometry in Computer-Aided Design.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view Convolutional Neural Networks for 3D Shape Recognition, in: Proceedings of the Ieee International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2015.114

Tangelder, J.W.H., Veltkamp, R.C., 2008. A survey of content based 3D shape retrieval methods. Multimed Tools Appl 39, 441–471. https://doi.org/10.1007/s11042-007-0181-0

Vidanes, G., Toal, D., Zhang, X., Keane, A., Gregory, J., Nunez, M., 2024. Extending Point-Based Deep Learning Approaches for Better Semantic Segmentation in CAD. CAD Computer Aided Design 166. https://doi.org/10.1016/j.cad.2023.103629

Willis, K.D.D., Jayaraman, P.K., Chu, H., Tian, Y., Li, Y., Grandi, D., Sanghi, A., Tran, L., Lambourne, J.G., Solar-Lezama, A., Matusik, W., 2022. JoinABLe: Learning Bottom-up Assembly of Parametric CAD Joints, in: Proceedings of the Ieee Computer Society Conference on Computer Vision and Pattern Recognition. pp. 15828–15839. https://doi.org/10.1109/CVPR52688.2022.01539

Wu, R., Xiao, C., Zheng, C., 2021. DeepCAD: A Deep Generative Network for Computer-Aided Design Models, in: Proceedings of the Ieee International Conference on Computer Vision. pp. 6752–6762. https://doi.org/10.1109/ICCV48922.2021.00670

Zhou, S., Tang, T., Zhou, B., 2023. CADParser: A Learning Approach of Sequence Modeling for B-Rep CAD, in: Ijcai International Joint Conference on Artificial Intelligence. pp. 1804–1812.

Contact: J. M. Grønvald, Technical University of Denmark, jamgr@dtu.dk