Classification of CAD-Models Based on Graph Structures and Machine Learning

Robin Roj[1], Maxim Sommer[2], Hans-Bernhard Woyand[3], Ralf Theiß[4] and Peter Dültgen[5]

[1]Forschungsgemeinschaft Werkzeuge und Werkstoffe e.V., roj@fgw.de
[2]University of Wuppertal, maxim.sommer@uni-wuppertal.de
[3]University of Wuppertal, woyand@uni-wuppertal.de
[4]Forschungsgemeinschaft Werkzeuge und Werkstoffe e.V., theiss@fgw.de
[5]Forschungsgemeinschaft Werkzeuge und Werkstoffe e.V., dueltgen@fgw.de

Corresponding author: Robin Roj, roj@fgw.de

Abstract. For the support of the designer during the construction process of new products, professional software solutions are commercially available. Beside the basic functionality for the creation of parts, assemblies, and technical drawings, also product data management, including features for teamwork or the convenient storage of large amounts of files, is provided. A daily routine is the revision and modification of already existing components. Such tasks require a search engine for CAD-models that compares two files and detects similarities. For the solution of problems like that, in this paper a method for the automated classification of CAD-models based on fingerprints in form of graph structures is enhanced. A machine learning approach is used to classify the models of a dataset consisting of more than 2000 files. It is shown that the method automatically creates clusters of parts with geometric similarity.

Keywords: CAD, classification, clustering, database, fingerprint, graph structure, machine learning, PDM
DOI: https://doi.org/10.14733/cadaps.2022.449-469

1 INTRODUCTION

Technical designs of new products are nowadays mainly developed using CAx-methods, that are no longer withheld from a minority of specialists. The factors of time saving, detection of error potential, and compliance with quality standards are becoming more and more important due to increasing market requirements and are the driving force behind the progress of those systems. Also, new technologies and capabilities in production, such as additive manufacturing or topology optimization, lead to an increase in complexity and effort. In combination with the sharp increase in available computing power, the amount of data generated by these systems is now conceivably high.

In the field of computer aided engineering (CAE) in general and computer aided design (CAD) specifically, these circumstances lead to challenges in an efficient usage. For a rapid and straightforward access to the CAD-database, it is remarkably important that the designer is able to access and overview all existing information. Usually, the search for similar or specific 3D-models that were created and saved in older projects is time consuming and error-prone. It can also lead to the risk of constructing CAD-models twice or several times, although a corresponding version is already existent in the collection. Modern systems for product data management (PDM) or product lifecycle management (PLM) promise remedy. Programs like that contain not only geometrical information, but also support the users with organization, manufacturing or accounting as well as with quality management or design of experiments. This illustrates that a structured and well-organized database is of great importance and that additional tools, like e.g., a search engine for CAD-models or an automated clustering algorithm, support productive and targeted work.

For the solution of problems like that, the here presented approach contains a transformation of CAD-files into graph structures in order to detect similarities and build clusters and combines this method with modern machine learning (ML) techniques by Sommer [1]. After the state of the art is formulated in section 2, the methodology in combination with the approach, the parameters of comparison, and the clustering is presented. Subsequently, some results illustrate the performance capabilities of the presented algorithms, and the outcome is discussed at the end of the paper.

2 STATE OF THE ART

As already mentioned in section 1, PDM-software represents a solution for organized teamwork. Vault by Autodesk offers a PDM-solution in which the information of the design and planning can be managed and tracked [2]. In particular, this involves information about the organization of the data creation, simulation, and documentation processes. In addition, the system also offers revision management functions. The company refers to a "vault" as a central storage location (database) where all files (not only geometrical data) are stored. This vault is accessible to any registered user and allows easy access to the data it contains. All file properties are also stored for quick search and retrieval. If an employee claims a file in the vault for editing and thus "checks out" of the vault, the file is reserved exclusively for the corresponding user for the duration of editing. Once editing is completed and the file is released again, all other users regain access to it. This ensures that only one person is working on a version at the same time and that the collaboration of the design team is successful. All versions and dependent files are also archived and bundled so that the history of product development is not lost and design steps that may have already been carried out are not repeated.

The PDM-system Helios from ISD has a similar structure and can be divided into the categories document management, CAD-data-management and product lifecycle management [3]. CAD-data-management is a processing method of data that runs centrally. This means that the creation or modification of CAD-data is not only visible locally to the user, but equally visible to all involved. With the direct linking of documents and article data, the cross-departmental information about a product can always be found in a bundled form. The PDM-system also offers a search function for CAD-models in which individual masks or specific attributes of the models can be searched. With version management, modified files are given a corresponding number for recognition. In addition, Helios also offers the classification of CAD-documents. The models are sorted by comparing the entries in attribute lists.

In addition to PDM-systems like that, also specialized software for the comparison of similarity is available. For example, classmate CAD is a patented software solution from Simus systems that can be used as a supplement to existing software solutions in companies [4]. At its core, classmate CAD offers fully automated geometric classification and indexing of 3D-models for similarity search. The assignment of CAD-models is made possible by matching the characteristics in the feature lists. For this purpose, the software solution determines the characteristics of the

features during the analysis of form elements of a CAD-model and independently adds them to the feature lists of the corresponding models. The geometrically based similarity search is realized by the calculation of a "geometric fingerprint" which is also referred to by the company as an "index". Based on this index, a similarity comparison is realized. Classmate CAD can be embedded in several common CAD-systems. Further similar solutions are Similia by Simuform [5], Geolus Shape Search by Siemens [6], and Exalead Onepart by Dassault Systèmes [7].

While systems like that are commercially available and already established, the topic of recognition and classification of CAD-models is still of high interest. In the following section, several scientific approaches are presented. One common attempt is the feature-based approach. While non-native file formats that are used for exchange (e.g., STEP, STL, IGES) contain only the geometrical information, the native files usually save the procedure that is executed by the designer. Thus, a model can be subdivided into many elements called features. Those can consist of sketch elements, like dots, lines, circles, or arcs, as well as three-dimensional components, like extrusions, drillings, chamfers, or sweepings. Shi et al. provide an overview of the most important research results, such as rule- and graph-based feature recognition, volume decomposition, and artificial neural network-based systems, and evaluate them critically [8]. The work of Al-wswasi and Ivanov is more specific and concentrates on an interactive feature recognition system for rotational parts using STEP-files [9]. While not only the recognition is a challenge in their opinion, also the connection to the manufacturing and process planning, respectively, is sophisticated. They point out the fact that the design is geometry based, whereas process planning is manufacturing feature based.

Mun and Kim present an algorithm for the simplification of CAD-models [10]. They consider an extended feature-based simplification method in which a multi-branch feature tree is determined using a feature dependency graph. Both additive and subtractive features are considered in order to evaluate the importance of each and delete the most unimportant ones.

As a consequence of the difficulty to detect and recognize features, it is one option to take the structural composition of the CAD-model already during the design process into account. Sun et al. propose a robust design method to prevent feature failure [11]. Thereby, the reuse of CAD-models in other projects and the further processing as well as changing is less error-prone. A similar idea is introduced by Li et al., where a new design method based on feature reusing of a non-standard cam structure is presented [12]. Also, Liu and Wang consider machining features and connect their approach with the process planning during manufacturing [13].

For the detection and comparison of CAD-models, the feature recognition is just the first step. In order to classify or cluster geometries, it is necessary to develop methods for the comparison of two files. A typical software functionality is a search engine, where e.g., a reference part can be specified by the user. Nasution describes the modelling and simulation of such a search engine [14]. Hilaga et al. propose the use of a multi-resolutional reeb graph in order to estimate the similarity of 3D-shapes and match topologies automatically [15]. Finally, Lupinetti et al. demonstrate that not only the comparison of single CAD-files is of great interest, but also the consideration of whole assemblies including their contained components can support the work of the designers permanently [16].

In recent years, modern technology in the field of ML and deep learning has been applied to clustering and classification problems of CAD parts. Qin et al. developed a method in which the features are selected and extracted from the CAD models first [17]. In the next step, high dimensional input vectors for their neural network are preprocessed for category recognition. As a result, with a total number of over 7000 models and 28 categories, Qin et al. reach an average correct rate of 98.64 % outperforming similar approaches. Also, Ip and Regli present an ML technique for a content-based classification [18]. In their fully automated shape-based approach, they reach an average correctness of 63 % and describe the potential to increase this value by improving the training data.

Jayanti et al. compare two shape-based clustering methods and analyze their effectiveness [19]. In the first one, the distances are transformed into feature spaces using k-Means clustering.

In the second one, the original distances with a distance-based clustering algorithm are directly used. With a benchmarking test set of 867 models, they compare both approaches. Also, Rucco et al. present a methodology for part classification with supervised ML [20]. In contrast to the others, they focus on the detection of features serving as input for an artificial neural network.

3 METHODOLOGY

In order to prove and demonstrate the approach, a CAD-database consisting of more than 2000 models has been created and automatically translated into the non-native VRML-format (Virtual Reality Modeling Language). Based on the complete geometrical information of every model, a fingerprint, which represents only the most important characteristics, has been created. This contains the examination of every surface and the categorization into different types. The visualization of the fingerprint containing only the reduced information can be depicted in form of a frame. There, the centroid of each surface is represented as a sphere that is connected with all respective other surfaces, leading to a three-dimensional graph structure.

The derivation of the fingerprints served as a preparational step to process and evaluate the data in order to sort and categorize the database in the next steps. For this automatic classification, a k means-clustering-algorithm is applied. Further features of the CAD-models, like e.g., the bounding box, the surface-ratio, the centroids, and the surface densities have been considered.

Conclusively, some detected clusters of the database are presented, and the effectiveness and performance capabilities of the algorithms is evaluated. It is shown that the whole method represents a powerful tool serving as a search engine for CAD-models.

3.1 Approach

The approach for extracting geometric properties of a CAD-model according to Roj forms the basis for the application of ML-methods and is therefore described in detail [21]. In essence, the approach can be abstracted as a procedure in which geometric information of CAD-models of any format can be reduced. This information is finally available in text format and can thus be used for geometric comparison. The approach can be divided into three sub-processes.

First, the CAD-model is exported from a random native format to the VRML format. The term "native" refers to a file format that has been developed by a company and is used only in the respective system. The VRML-format is practical because of the unique indexing of geometry elements. After analyzing the geometry of the VRML-model, only data that provides information about the surface structure is extracted. Constructed is the surface of more complex geometric elements predominantly with sets of lines and two-dimensional surfaces, which have in turn a Cartesian definition by vertices. Sections with corresponding information are found and interpreted by an algorithm. In Table 1 an overview is provided, showing which color code corresponds to which surface type.

Color	Type	Definition
	round	round surfaces
	circle	circular surfaces
	plane	surfaces in one plane
	tri	triangular surfaces
	four	quadrangular surfaces
	six	hexagonal surfaces
	free	freeform surfaces

Table 1: Color code of surface types.

Surfaces consisting of three, four, or six vertices are identified as triangular, quadrilateral or hexagonal. For more complex point constellations, it is investigated whether the resulting surface can be identified as circular, has radial elements, lies on a plane, or can be understood as a freeform surface. For marking purposes, each identification in the VRML source code is given a color designation. Thus, the VRML-model appears with differently colored surfaces, depending on the interpreted type. For consideration, Figure 1 shows two exemplarily colored VRML-models. On the left a hexagon head screw is depicted. There, the coloring approach of round elements, like shafts and even drillings, becomes clear. While also the contact face of the head is colored in yellow, it can be seen that the bottom is blue, due to its circular shape, and the lateral faces are detected as freeform shapes. On the right side a more complex part that is used as a mounting device is depicted. Especially with the grey and cyan areas it is shown that the coloring is not always obvious at first sight.

For the creation of a fingerprint, all surfaces are extracted and recorded in a text file to isolate the crucial information and allow easy further processing. Every line of the text file represents one surface. Figure 2 shows the text-based fingerprint of the mounting device. First, the designation of the surface type appears (colored according to Table 1). This is followed by Cartesian position information of the area center on the x-, y- and z-axis, related to the original coordinate system (dark red). Subsequently, it is indicated to which other surface centers a connection exists (light green) and how large the distances are (dark blue). This subsequent information allows conclusions to be drawn about the relationship between the surfaces of the CAD-model in its original state.

Additionally, it is useful to visualize the fingerprint. In three-dimensional space, the surface centers are displayed according to their position coordinates and information regarding the surface type and number as well as distance of the connections. In Figure 3 the resulting graph structure of the visualized fingerprint of the mounting device is depicted from two different angles. The representation is helpful because it allows to see to what extent the fingerprint differs geometrically from the original CAD-model. Each surface is represented by a respectively colored sphere. Also, the connections of all surfaces are indicated by gray rods. For a better overview the two perspectives have been chosen by just slightly turning the same graph in order to show the other side.

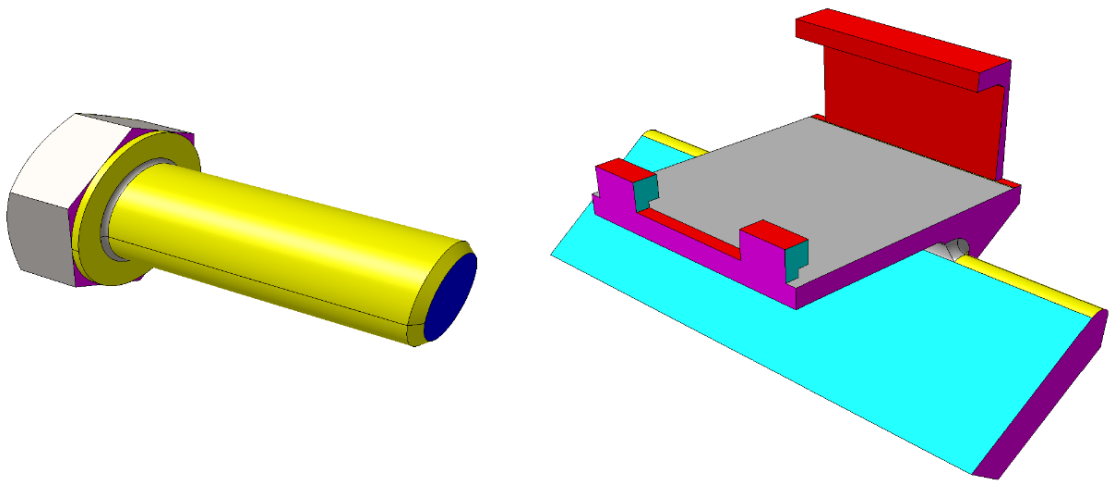


Figure 1: Colored VRML-files of a screw and a mounting device.

00['four', -36.87, 0.0, 56.41, 30, 6.65, 37, 42.17, 1, 6.44, 41, 42.17]
01['four', -31.87, 0.0, 60.48, 0, 6.44, 37, 44.88, 39, 8.53, 41, 44.88]
02['round', -46.62, 0.0, 10.12, 41, 46.79, 38, 22.01, 37, 46.79, 3, 0.27, 6, 41.28, 42, 41.28]
03['round', -46.5, 0.0, 9.87, 5, 66.58, 6, 41.29, 7, 48.37, 4, 66.58, 42, 41.29, 2, 0.27, 41, 46.92, 37, 46.92]
04['plane', -0.29, 45.0, -6.64, 40, 76.90, 42, 51.01, 25, 55.20, 14, 27.79, 12, 22.53, 13, 16.80, 19, 46.76, 29, 49.88, 49, 70.14, 7, 48.02, 3, 66.58]
05['plane', -0.29, -45.0, -6.64, 7, 48.02, 49, 70.14, 29, 49.88, 19, 46.76, 9, 16.80, 8, 22.53, 11, 27.79, 25, 55.20, 40, 76.90, 6, 51.01, 3, 66.58]
06['four', -48.37, -41.25, 10.0, 5, 51.01, 40, 47.14, 37, 28.76, 2, 41.28, 3, 41.29]
07['free', 1.87, 0.0, 10.0, 4, 48.02, 49, 48.18, 34, 60.33, 33, 43.17, 47, 60.33, 5, 48.02, 3, 48.37]
08['free', -22.67, -47.5, -7.45, 5, 22.53, 11, 12.33, 10, 12.74, 23, 27.70, 24, 30.31, 9, 6.19, 22, 28.32]
09['free', -16.86, -47.5, -5.30, 22, 27.62, 20, 47.59, 21, 61.36, 19, 47.90, 5, 16.80, 8, 6.19, 23, 28.05]
10['round', -22.44, -48.23, -20.17, 25, 51.03, 24, 26.77, 8, 12.74, 23, 28.81, 11, 3.61]
11['round', -24.94, -45.73, -19.44, 8, 12.33, 5, 27.79, 25, 49.35, 10, 3.61]
12['free', -22.67, 47.5, -7.45, 18, 27.70, 15, 12.74, 16, 30.31, 14, 12.33, 4, 22.53, 13, 6.19, 17, 28.32]
13['free', -16.86, 47.5, -5.30, 4, 16.80, 19, 47.90, 20, 47.59, 17, 27.62, 21, 61.36, 12, 6.19, 18, 28.05]
14['round', -24.94, 45.73, -19.44, 25, 49.35, 4, 27.79, 12, 12.33, 15, 3.61]
15['round', -22.44, 48.23, -20.17, 12, 12.74, 16, 26.77, 18, 28.81, 25, 51.03, 14, 3.61]
16['four', -21.71, 75.0, -20.17, 12, 30.31, 15, 26.77, 18, 10.53, 28, 42.13, 25, 76.76]
17['round', -15.90, 75.0, -7.80, 28, 45.78, 21, 83.38, 13, 27.62, 20, 75.05, 12, 28.32, 18, 4.69]
18['round', -20.17, 75.0, -9.76, 12, 27.70, 15, 28.81, 16, 10.53, 28, 46.78, 17, 4.69, 13, 28.05]
19['free', -11.95, 0.0, -1.54, 13, 47.90, 4, 46.76, 29, 32.16, 5, 46.76, 9, 47.90, 20, 4.69]
20['free', -13.91, 0.0, -5.81, 9, 47.59, 21, 36.43, 22, 75.05, 13, 47.59, 17, 75.05, 19, 4.69]
21['six', 10.83, 0.0, -32.55, 17, 83.38, 28, 100.29, 27, 35.40, 26, 100.29, 22, 83.38, 9, 61.36, 20, 36.43, 13, 61.36]
22['round', -15.90, -75.0, -7.80, 9, 27.62, 20, 75.05, 21, 83.38, 26, 45.78, 23, 4.69, 8, 28.32]
23['round', -20.17, -75.0, -9.76, 26, 46.78, 24, 10.53, 8, 27.70, 10, 28.81, 9, 28.05, 22, 4.69]
24['four', -21.71, -75.0, -20.17, 10, 26.77, 25, 76.76, 26, 42.13, 23, 10.53, 8, 30.31]
25['plane', -14.93, 0.0, -35.06, 26, 102.35, 27, 51.39, 28, 102.35, 16, 76.76, 15, 51.03, 14, 49.35, 4, 55.20, 40, 75.76, 5, 55.20, 11, 49.35, 10, 51.03, 24, 76.76]
26['plane', 6.56, -100.0, -38.89, 21, 100.29, 27, 105.17, 25, 102.35, 24, 42.13, 23, 46.78, 22, 45.78]
27['four', 27.78, 0.0, -63.63, 21, 35.40, 26, 105.17, 28, 105.17, 25, 51.39]
28['plane', 6.56, 100.0, -38.89, 16, 42.13, 25, 102.35, 27, 105.17, 21, 100.29, 17, 45.78, 18, 46.78]
29['four', 20.17, 0.0, 0.0, 4, 49.88, 19, 32.16, 49, 32.33, 5, 49.88]
30['four', -43.42, 0.0, 55.20, 37, 41.52, 38, 23.19, 0, 6.65, 41, 41.52]
31['four', 43.0, 32.0, 14.0, 36, 5.85, 44, 10.16, 34, 9.97, 33, 32.12]
32['four', 43.0, -32.0, 14.0, 33, 32.12, 35, 10.16, 47, 9.97, 48, 5.85]
33['plane', 45.0, 0.0, 12.0, 35, 23.77, 45, 2.69, 44, 23.77, 31, 32.12, 34, 41.37, 7, 43.17, 47, 41.37, 32, 32.12]
34['six', 45.5, 41.0, 17.5, 7, 60.33, 49, 41.54, 43, 11.71, 36, 10.25, 31, 9.97, 33, 41.37]
35['six', 45.5, -23.0, 18.0, 45, 24.12, 49, 24.07, 46, 11.40, 48, 10.17, 32, 10.16, 33, 23.77]
36['four', 41.0, 32.0, 19.5, 34, 10.25, 43, 7.10, 44, 10.17, 31, 5.85]
37['plane', -40.93, -37.5, 37.53, 6, 28.76, 40, 38.55, 39, 46.40, 1, 44.88, 0, 42.17, 30, 41.52, 38, 38.18, 2, 46.79, 3, 46.92]
38['four', -45.72, 0.0, 32.12, 30, 23.19, 37, 38.18, 41, 38.18, 2, 22.01]
39['four', -39.22, 0.0, 64.81, 1, 8.53, 37, 46.40, 40, 33.32, 41, 46.40]
40['free', -48.58, 0.0, 32.82, 6, 47.14, 37, 38.55, 5, 76.90, 25, 75.76, 4, 76.90, 42, 47.14, 41, 38.55, 39, 33.32]
41['plane', -40.93, 37.5, 37.53, 30, 41.52, 38, 38.18, 0, 42.17, 1, 44.88, 39, 46.40, 40, 38.55, 42, 28.76, 2, 46.79, 3, 46.92]
42['four', -48.37, 41.25, 10.0, 40, 47.14, 41, 28.76, 4, 51.01, 3, 41.29, 2, 41.28]
43['four', 45.5, 32.0, 25.0, 36, 7.10, 44, 11.40, 34, 11.71, 49, 34.64]
44['six', 45.5, 23.0, 18.0, 31, 10.16, 36, 10.17, 43, 11.40, 49, 24.07, 45, 24.12, 33, 23.77]
45['four', 47.5, 0.0, 11.0, 44, 24.12, 49, 2.91, 35, 24.12, 33, 2.69]
46['four', 45.5, -32.0, 25.0, 47, 11.71, 48, 7.10, 35, 11.40, 49, 34.64]
47['six', 45.5, -41.0, 17.5, 32, 9.97, 48, 10.25, 46, 11.71, 49, 41.54, 7, 60.33, 33, 41.37]
48['four', 41.0, -32.0, 19.5, 35, 10.17, 46, 7.10, 47, 10.25, 32, 5.85]
49['plane', 50.0, 0.0, 12.5, 5, 70.14, 7, 48.18, 47, 41.54, 46, 34.64, 35, 24.07, 45, 2.91, 44, 24.07, 43, 34.64, 34, 41.54, 4, 70.14, 29, 32.33]

Figure 2: Text-based fingerprint of the mounting device.

The fact that the CAD-fingerprint is stored in a text file is advantageous for the handling of large databases. The information is easily accessible and can be viewed quickly without the need for special applications. This circumstance consequently facilitates the further processing of the data by different applications. In addition, it can also be noted that working with text files in programming languages is particularly favored by the good documentation about it, which is

beneficial for programming. Furthermore, the stored fingerprint contains information that is well suited for shape comparisons. On the one hand, proportions of the model can be derived from the information on adjacent surfaces and their distances. On the other hand, the assignment of data records to the respective surface types enables conclusions to be drawn about their relationship.

This information can be used directly as criteria in the shape comparison by any program. Also, worth mentioning is the manageable amount of information associated with text files. Instead of extracting data sets from many points that define the component in its entirety, only information about the resulting surfaces is listed. Since the recognition of surfaces is well understood from a human point of view to describe the object under consideration, this approach already provides a helpful first step towards the comparison of 3D-shapes. Also, the simplification leans heavily on the original representation of the model since the positions of the individual surface centers correspond to those in the original representation of the model.

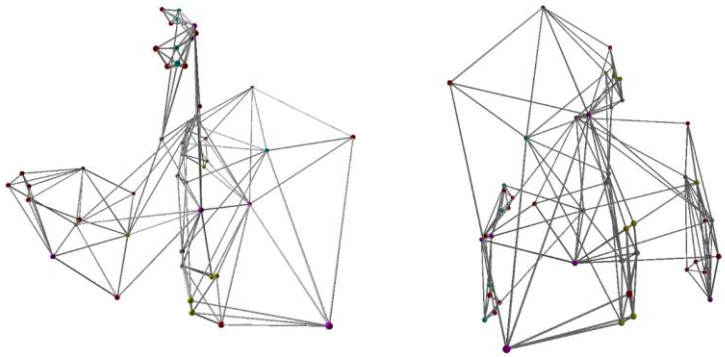


Figure 3: Visualized graph of the mounting device in two different angles.

3.2 Parameters of Comparison

Since an ML-method can only make assignments or predictions if it draws on experience, a knowledge base is the most important prerequisite. The database used in this approach contains more than 2000 arbitrary CAD-models in VRML-format. Since they were randomly collected, the parts are not presorted into certain categories. Due to a high geometrical variety, it was expected that only few clusters (e.g., standard parts) arise. The extent of this data collection determines the variety of experiences. The more experiences are available, the more accurate the assignment or prediction becomes. The data collection was built from CAD-models of technical components. In contrast to assemblies, components are not based on links of single components and can therefore be regarded as elementary. This has the advantage that the complexity of the data to be processed is lower and thus comparison mechanisms can potentially be applied faster.

The choice of the output format of the CAD-data for the ML-application is based on a number of prerequisites. First, the format should preferably contain only information that is relevant to the application. Redundant content takes up more storage capacity and thus increases processing time. Since the classification of CAD-models on the basis of geometric characteristic is dealt with here, information about the geometry has to be included. Information, such as material designation or costs, is not target-oriented and is therefore disregarded. On the other hand, the format should be up to date and recognized by common CAD-applications for processing purposes. This ensures the compatibility of the future ML-application since deviating CAD-formats can, in most cases, be converted into the required VRML output format. The character encoding of the respective format is also important. The content should be in ASCII-format for readability since understanding the structure and meaning of the records is crucial to comprehend the relevance of

the entries. The above conditions were chosen to simplify the development and testing of the ML-model, but the results can be applied to all databases afterwards.

An effective basic principle for comparing elements is based on defining metric relations between them. From a mathematical point of view, a metric is the distance between two points in space. If the space under consideration contains any number of such points, these distances can be used to mathematically determine how similar the respective points are to each other. The transfer of this principle to the similarity comparison of CAD-models can be simplified by reducing corresponding models to fingerprints. Since previously mentioned advantages favor the approach for the generation of fingerprints, it is considered as a basis for all following methods. For the implementation of the comparisons between the fingerprints, the representation of these in the form of data points is suitable. In the following, these data points are defined by certain entries, which together form a vector. For example, if the space is three-dimensional, the position of a data point is defined by a vector with exactly three entries. In an n-dimensional space, the data point is represented by a vector with n entries. The definition of these entries for the comparison allows conclusions to be drawn about the relationship between the original CAD-models. If two text-based fingerprints (like e.g., in Figure 2) should be compared with each other, it might be difficult due to the fact that the number of surfaces is different. To solve this, for the application of ML, each text-based fingerprint needs to be broken down into 18 parameters of comparison. Those can be considered as the hyper parameters for the ML algorithms used for classification and clustering. Table 2 shows the 18 parameters of comparison exemplarily for the mounting device from Figure 1-3.

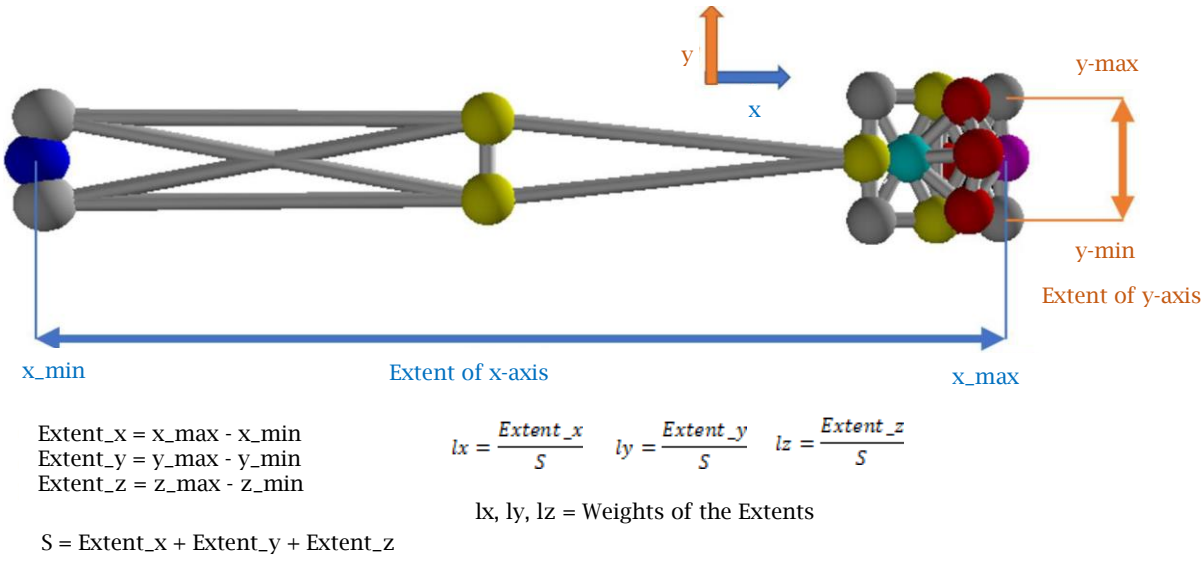


Figure 4: Weight of lengths l1, l2, and l3 for one fingerprint.

The entries of each data point can be understood as expressions of certain characteristics related to the fingerprint of the 3D-model. When defining parameters of comparison, three circumstances have to be taken into account, which can influence the comparison to a particular extent. First, it must be possible to determine the parameters independently of the scaling of the fingerprint. Even if the shape of two comparison objects is identical, the comparison may still fail due to different scaling of the two original CAD-models. Secondly, it must be taken into account that the coordinates of the points of the fingerprint always refer to the local coordinate system. The positions of the coordinate systems of the comparison objects may differ, depending on the construction process of the 3D-model. The third circumstance refers to the orientation of the object in space. E.g., the comparison of bolts may fail if the objects of comparison are similar but

point in different directions. With respect to these circumstances, the parameters are presented in Figure 4-6 that allow for a suitable comparison.

A bounding box is a group of parameters that provides information about the proportions of the fingerprint. The parameters are length ratios of the fingerprint on each of the three spatial axes. To determine these values, the lines of the fingerprints are first searched for position coordinates in text form. Each line corresponds to the spatial center of a surface of the 3D-model and contains information about the position on the x-, y- and z-axis in space. For n-surfaces, which are entered in the fingerprint, n-coordinates per axis are found. From the n-coordinates of an axis, the maximum and minimum values are extracted, the difference of which represents the extent of the fingerprint in the affected axis. These differences are obtained from all three axes, which results in three dimensions.

If these values are compared with each other, statements can already be made as to whether this is a CAD-model that is more elongated or uniform. However, since these are differences that depend decisively on the scaling of the original model, a further step must be taken. Instead of working with absolute values, the respective dimensions are divided by the sum of all three. This gives a weighting and thus the desired independence from the scaling. Figure 4 illustrates the determination of the comparison characteristics.

It is important to remember that the fingerprints can be positioned differently in space, according to the CAD-models. Strictly comparing the weighting of the x-axis of one fingerprint with the weighting of the same axis of another fingerprint is thus not reasonable. The problem is solved by a fixed ranking order in which the weights are sorted. The label l1 always contains the largest weighting. The second largest is defined as l2 and the smallest weighting can be found as l3. If the bounding box of two fingerprints is compared, l1 is always related to l1, l2 to l2 and l3 to l3, regardless of which axis these ratios refer to. The characteristics of a bounding box can each have a value between 0 and 1, where 1 is the maximum.

The surface ratios of a fingerprint can also be determined from the information of the lines. Each line in the fingerprint contains a designation about the surface type corresponding to a surface of the CAD-model. Since there are seven possible designations, the same number of parameters can be derived. The extraction of these parameters can be divided into two steps.

In the first step, the fingerprint is searched for corresponding labels. In concrete terms, this involves designations, such as "circle", "round" or "plane". Each of the terms is assigned the number of times it occurs. Since the amount of surfaces that appear in each fingerprint sometimes varies considerably, the second step is to generate ratios. If the existing number of surfaces of a specified type is divided by the total number of surfaces of the fingerprint, the desired ratio is computed. This procedure is performed for all labels that can be found in the rows of the fingerprint.

The results are again weightings, which are in the number range from 0 to 1. The designation "ratio_round", e.g., contains the proportion of the area type "round". The same applies to the parameters "ratio_circle", "ratio_plane", "ratio_tri", "ratio_four", "ratio_six", and "ratio_free". If a CAD-model has a rather round shape (like e.g., a rotationally symmetric turned part), the value for the "ratio_round" is comparatively higher than the other ratios. Thus, it is an indicator of the percentage and as it can be seen in Table 1, the sum of all ratios is always 1. Further important characteristic values result from the center of gravity ratio. This represents three parameters that are related to the spatial center of gravity of the fingerprint. Since the individual area centers in the fingerprint can be represented by points in three-dimensional space, their centroids can be described by three average values of the coordinates on each axis. The feature extraction consists of three parts.

First, all position values of the data points in the fingerprint are acquired for each axis respectively, directly followed by the formation of the arithmetic mean of the totality of all values for each axis. As a result, mean values are obtained for each of the three axes, which combined result in the centroid of the fingerprint.

Next, the relation of a mean value to the respective extent is established to solve the dependence on the scale of the CAD-model. For a more detailed understanding, Figure 5 can be considered. There, this aspect is exemplarily demonstrated on the x-axis of a screw. According to the equations, the values for y- and z-axis can be computed respectively. To obtain the desired relation, the distance between the mean value (B) and the smallest value (A) is determined, which in turn is divided by the extent of the affected axis. In the example A is the smallest and C is the largest x-value of all surface centers. B represents the mean value, taking all surfaces into account and calculating the average value. In case of the screw, it is comprehensible that B is placed on the right-hand side since the screw head consists of many surfaces. For the calculation of the ratios c1, c2, and c3, the distances between the smallest values (A) and the mean values (B) are divided by the complete span along the respective axis using the largest value (C). This leads to the equation of AB/AC.

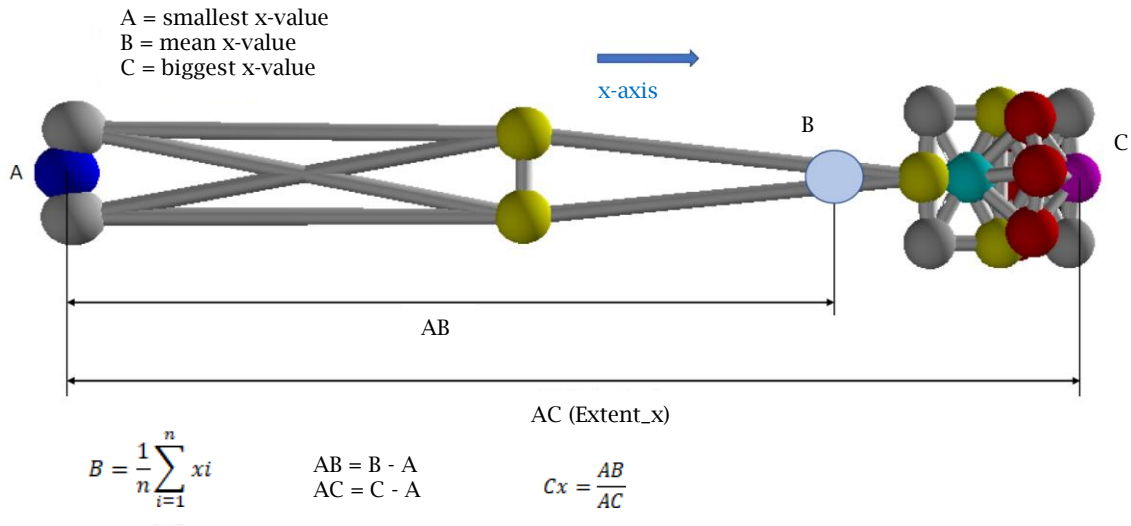


Figure 5: Ratios of average extents c1, c2, and c3.

Finally, it is ensured that the comparison of these ratios can be undertaken regardless of the position of the 3D-model in space. For this purpose, the ratios are sorted in a ranking order considering the largest, the medium and the smallest span of the model, identical to the procedure for the bounding box. The largest weighting is found under c1. Consequently, c2 is assigned the second largest ratio, while c3 represents the smallest ratio. The results here also range within the values between 0 and 1. With the help of these parameters, the localization of the center of gravity of a fingerprint, and thus approximately that of the CAD-model, is to be quantified.

With the density ratio, parameters are presented that relate to the number of surface centers in specific areas of the visual fingerprint. As in the procedures before, the parameters represent individual ratios. Here, the concentrations of surface centers of individual surfaces are related to the total number of all centers. Again, the extraction of characteristic values can be divided into three steps.

In the first step, the axis of the fingerprint is identified on which the object has the largest extent. Under the bounding box, the largest extent is indirectly determined by the label l1. From information about the calculation of the weight l1, the relevant axis can be found. Subsequently, the visual fingerprint is fragmented into three equal parts based on its extent on the relevant axis. The number of parts is chosen on a trial basis and should be kept as simple as possible at first.

In the second step, for each part of the spatial fingerprint, the number of centers that can be found in it is recorded. The affiliation of these points to certain areas is defined by their position on the corresponding axis.

Finally, weights are also calculated from these values. The number of points of the individual subareas of the fingerprint are each divided by the total sum of all existing data points. The result is parameter b1 with the highest ratio and parameter b2 with the lowest ratio. These two parameters are each weight-determined at the two outer parts of the spatial fingerprint. Regardless of the size of these two parameters, the result is b_mid, which always denotes the weighting in the middle part of the fingerprint. Figure 6 illustrates the principle.

As a result of this approach, the parameters b1 are always compared with b1, b2 with b2 and b_mid with b_mid of the two affected fingerprints of the 3D-models. The introduction of a size ranking between b1 and b2 ensures here again the correct similarity comparison since the characteristic values depend no longer on the spatial position of the respective fingerprint and thus the 3D-model. The parameter b_mid always refers to the middle part of the affected fingerprint and is therefore independent of the spatial position of the model from the outset. With the combination of these characteristics, the concentration of surfaces at certain areas of the CAD-model is to be used for comparison. The values of the parameters are also in the numerical range between the minimum 0 and the maximum 1.

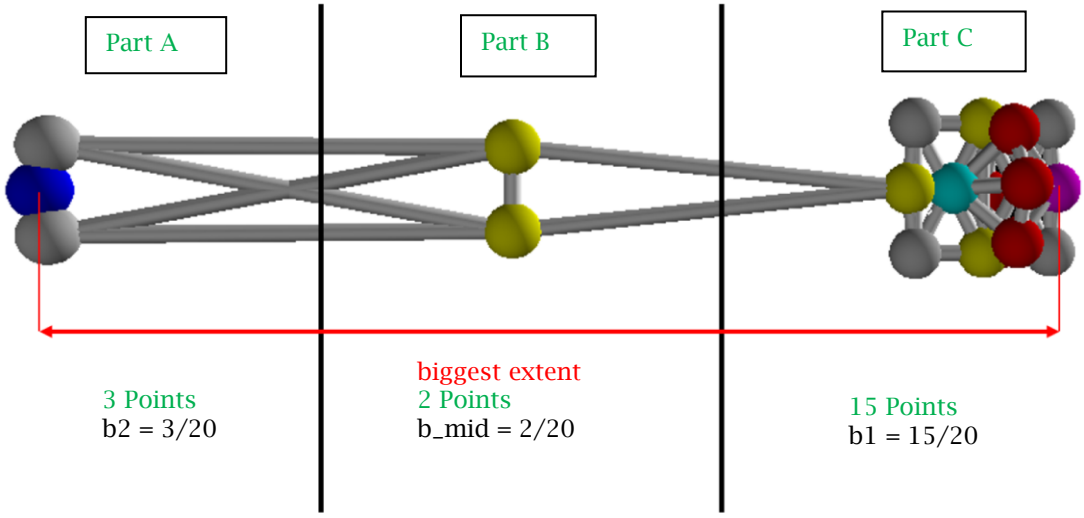


Figure 6: Ratios of density b1, b_mid, and b2 of a fragmented fingerprint.

Next, the idea of quantifying a surface that appears particularly dominant when viewing a 3D-model is described in detail. In the case of a screw, the cylindrical body that extends under the screw head is typical. In the fingerprint, this body is defined by two half cylindrical surfaces. The span can be used as an indicator for the dominance of these surfaces. The extraction of the spatial center of this surface from the fingerprint, based on the mentioned indicator, is accomplished in three phases.

Since each spatial center of area in the fingerprint is equipped with information about distances to other centers, these can be used to determine the respective span. To do this, the average value is first determined from all the distances that the data point in question has. This procedure is undertaken for each point that can be found in the fingerprint, thus obtaining as many average distances.

The dominant area is determined by comparing the average distances of each data point. The center with the largest average distance and the area it represents are assumed to be dominant. Finally, the internal label of the corresponding entry in the fingerprint is extracted and set as the parameter of the affected model. The labels are expressions such as "circle", "round", "four", etc., and a numerical value is taken in its place for simplicity. The type of the dominance surface is stored under the designation "main surface". In the following the assignment of numbers to the respective designations in the fingerprint are listed to: "circle" = 0, "round" = 1, "four" = 2, "plane" = 3, "free" = 4, "six" = 5, and "tri" = 6.

The extraction of the presented parameters of comparison from the fingerprints can be automated with the help of an appropriate algorithm. The operations described in the program for extracting the parameters are entirely based on the methods mentioned above. The goal of the code is to use automatic iterations through the fingerprint collection to extract all the parameters of comparison in one pass and merge them into one list.

The resulting list is structured by rows, which in turn is divided by columns. The rows are organized in a natural number sequence, starting with one, which allows the list to be indexed. Depending on how many fingerprints are captured by the program, there is also a corresponding number of numbered rows. The columns of the listing consist of the name of the respective fingerprint and the respective comparison characteristics. For a more detailed understanding, Table 2 can be considered. It shows the parameters of comparison for the mounting device from Figure 1.

Parameter	*Value*
ratio_plane	0.18
ratio_circle	0.0
ratio_tri	0.0
ratio_round	0.2
ratio_free	0.16
ratio_four	0.36
ratio_six	0.1
l1	0.468
l2	0.301
l3	0.231
c1	0.5
c2	0.533
c3	0.454
b1	0.24
b2	0.24
b_mid	0.52
main surface	2.0
main surface ratio	0.043

Table 2: Parameters of comparison for the mounting device.

3.3 Clustering

A prerequisite for the application of the k-Means algorithm to the data collection of the parameters of comparison is the automatic usage of existing data. In the following sections mainly the results from analyzing, processing, and visualizing data are shown.

A measure for assessing the clustering result is expressed via the "Silhouette Coefficient". Figure 7 contains the corresponding formula for obtaining the measured value as well as a sketch illustrating the procedure. Three clusters can be seen, which are indicated by the letters A, B, and

C. Each of the clusters contains points that are representative of individual data elements. For the procedure, a random data point is chosen, which is denoted by *i* in the figure. Based on the Euclidean distance, the distances of the point *i* to all other points within the same cluster are now calculated. The average value of all distances within the cluster, in this case cluster A, is represented by the variable *a*. Behind the variable *b* is the average value of the distances between the point *i* and all points of the neighboring cluster. In the figure, cluster B is located closest. Compared to C, B is located closer to A since the shorter average distance is determined here. By using the expression *max(a,b)*, the larger of the two average distances *a* and *b* is taken. Dividing the values according to the formula gives the Silhouette Coefficient *s* for point *i*. This process is repeated for each element in the dataset until a coefficient is assigned to each dataset. The coefficient can take a value between -1 and 1. If the coefficient of a data point is -1, then it is an element that has been assigned to an incorrect cluster. If the value 0 is taken, then the two clusters, which are compared by the formula, are identical. A value of 1 means that the corresponding point is infinitely far away from the neighboring cluster and thus represents the ideal case to be aimed at, since the respective clusters can be ideally separated from each other in this way.

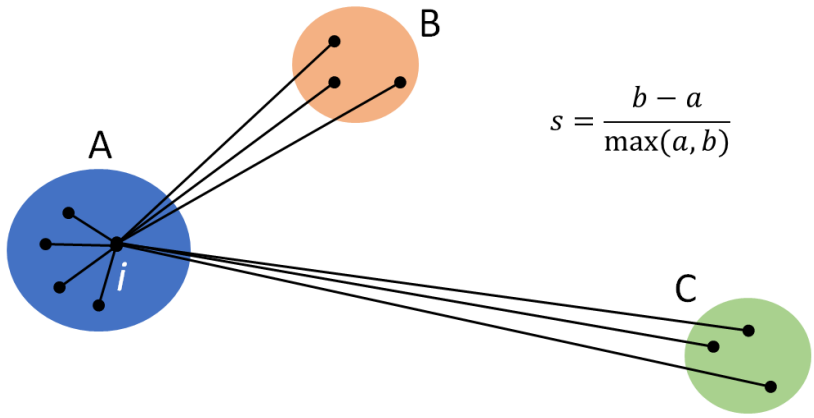


Figure 7: Silhouette Coefficient.

To start clustering, the k-means algorithm requires a specification for k. The designation k stands for the number of clusters into which the existing data set is to be divided. To get an idea of what value should be optimally chosen for k, the evaluation criteria for different values for k were calculated. In a loop, the coefficients for each k-Means model were determined, with the model that produced the highest coefficient being output at the end. Iteratively, the coefficients were determined for values of k between 1 and 1000. Since the clustering set had a size of over 2000 elements, the choice of interval was sufficient. The calculations to determine the optimal k-Means model showed that the model with the Silhouette Coefficient of around 0.341 represented the highest value and thus provided the best result. Thus, the data of the models were divided into 483 clusters. It is also worth mentioning at this point that CAD models are not assigned twice during the experiments and can thus only be found in a single cluster.

In order to get a better idea of the influence of the characteristics on the data set, graphical representations are used. In the following, the data set is therefore presented as a function of certain characteristics in scatter plots. A scatter plot is the representation of data from a data set as a function of two characteristics. The individual elements of the data set are plotted in a two-dimensional, Cartesian coordinate system. The values of a characteristic are plotted along each of the two axes. Depending on how high the values of the data points are, they are positioned accordingly in the coordinate system. This data element can be mapped symbolically, e.g. as a

point. The mapping of all elements of the data set in a coordinate system leads to an accumulation of points, which means that the resulting constellation can also be called a point cloud. The analysis of the shape and scatter of such a point cloud allows conclusions to be drawn about the behavior of the data set as a function of the parameter concerned.

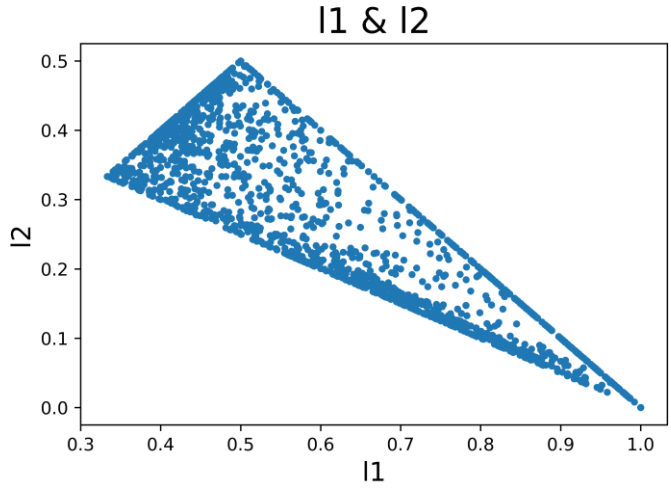


Figure 8: Formation of limits through dependencies (l1 & l2).

In the context of this paper, scatter plots were used to assess whether a particular parameter of comparison is suitable as a distinguishing criterion between elements. For the composition of a scatter diagram, comparison characteristics were used. It should be noted that these parameters are assigned to specific parameter groups. Parameters of comparison with the designations l1, l2 and l3 belong to the group bounding box, while the characteristics c1, c2 and c3 are assigned to the group center of gravity ratio. Such correlations hold for the remaining parameters as well. This fact matters because each of the following scatter plots were generated using only the pairs of parameters that are within the same parameter group. This approach aimed to capture the extent to which each characteristic is represented and the extent to which dependence on characteristics in the same group is apparent. Each blue dot represents an element of the dataset.

The following sections take up the specifics of the most interesting graphics and present an attempt to unlock the reasons behind them. Looking at the graphs, it can be seen that the compositions of the elements tend to have certain shapes, depending on the group membership of the parameters of comparison. Figure 8 shows the graph "l1 & l2", in which the scatter of elements seems to form a strict triangular shape. It is noteworthy that a solid boundary appears on all sides of the point cloud. This phenomenon can be explained by the relationship of the two parameters l1 and l2 to each other. These belong to the same parameter group and depend on each other due to their function. The parameters are designed so that each of them represents a proportion of the total length. If l1 takes a certain share of the total length, the values l2 and l3 can only take the remaining share of the length.

Figure 9 shows the dispersion taking into account the characteristics l2 and l3. The same situation arises since l3 also belongs to the bounding box group. However, it is noticeable that a long series of points show the expression 0.00 for the characteristic l3. This value means that l3 has no share in the total length.

From this it can be concluded that the fingerprints, from which the corresponding expression is extracted, extend only in two directions. In the third direction, the fingerprint has no extension and can therefore be described as two-dimensional.

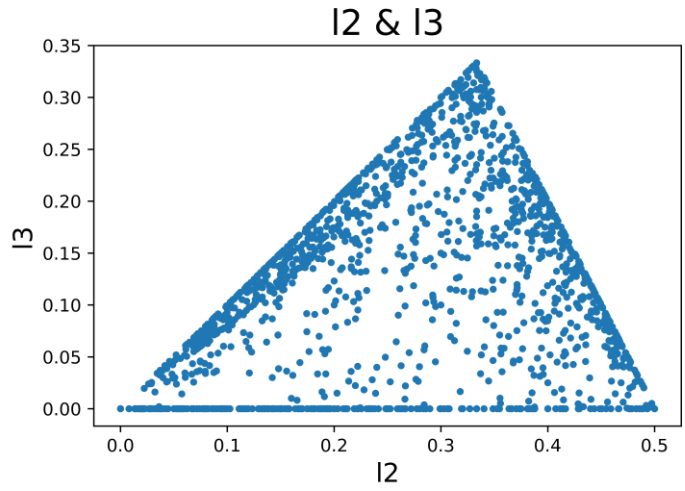


Figure 9: Aggregation of zero-values for l3 (l2 & l3).

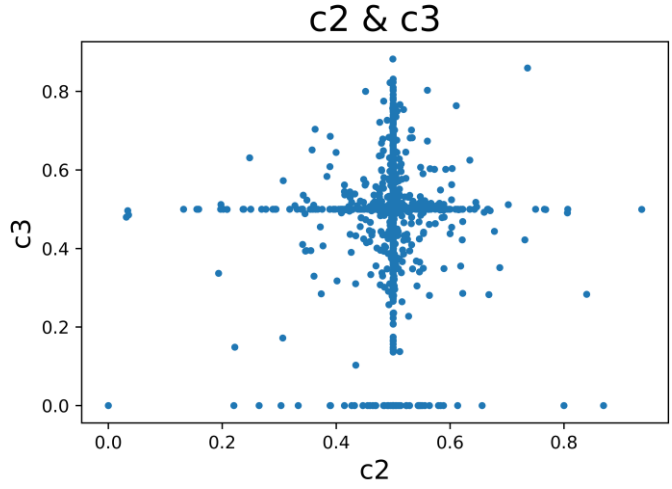


Figure 10: Aggregation of zero values for c3 (c2 & c3).

Zero values occur predominantly with the parameter l3 since this represents in principle always the smallest portion. In fact, the collection has some CAD-models whose spatially visualized fingerprints appear only two-dimensional.

Interesting trends can also be observed in the next group of characteristics. The parameters of comparison c2 and c3 belong to the "centroid ratio" group. The formation of a cross can be seen. In general, the majority of the data is located in the center of the diagram. If we look at the scales on the two axes, we see that very many elements line up at the value 0.5. This is true for the values of the characteristic c2 as well as for those of the characteristic c3. The value 0.5 means that the corresponding component of the center of gravity is located in the center of the respective dimension. The formation of a cross shape thus allows the conclusion that many CAD-models have centroid components that are centrally located in relation to the respective extent. These

constellations are conceivable for CAD-models that are designed symmetrically. The center of gravity of an object symmetrical to all directions always lies in the center.

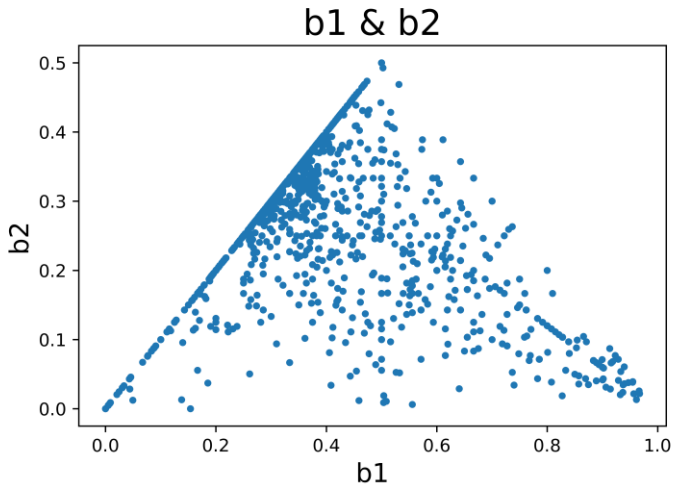


Figure 11: Linear relation between b1 and b2.

Figure 10 shows the data set in relation to the parameters c2 and c3. The red marking illustrates the fact that many elements have the value 0.0 for c3. This peculiarity can also be explained by two-dimensional fingerprints, which have no extension in the third direction and the affected centroid component is therefore 0.0.

Another peculiarity can be observed in the stagnation diagrams for the characteristics of the "density ratio" group. Figure 11 shows the diagram "b1 & b2", in which the peculiarity can be found within the red marking. It can be seen that a large number of elements line up within the mark. What is created here is a reflection of the values on both axes. If the expression of b1 of an element is the value 0.2, this also results for b2. It is interesting that this behavior does not apply to all elements in the data set. This phenomenon can be explained by the geometric symmetry of many CAD-models. The parameters of comparison b1, b2 and b_mid each represent one of the three parts over the longest extent of the fingerprint and each provide a ratio of the set of local surface centers to the total set. If the CAD-model is symmetrical, the two outer parts have an identical set of area centers from the extent. Also, this linear arrangement appears like a boundary. There are no elements located outside the boundary. The reason for this is the same as in the diagrams for parameters of the bounding box parameter group. In principle, the parameter b2 can only take values that are smaller than or at most as high as those of b1. If data points were found to the left of the separation, this would mean that there are values for b2 that are higher than b1.

Figure 12 presents the scatter plot "b2 & b_mid" in which this relationship is highlighted in red. However, another linear dependency is also conspicuous, which is emphasized by a green marking. Here, some data points seem to line up. To the extent that b2 decreases, the parameter b_mid appears to increase. This proportionality is only possible if the expression of the third parameter b1 remains constant in the cases in question. From this we can conclude that we are dealing with CAD-models which are basically very similar, but which certainly vary in a certain subrange. This subrange is captured accordingly by b2. One can imagine e.g. screws, which are generally very similar, but whose head area can vary geometrically.

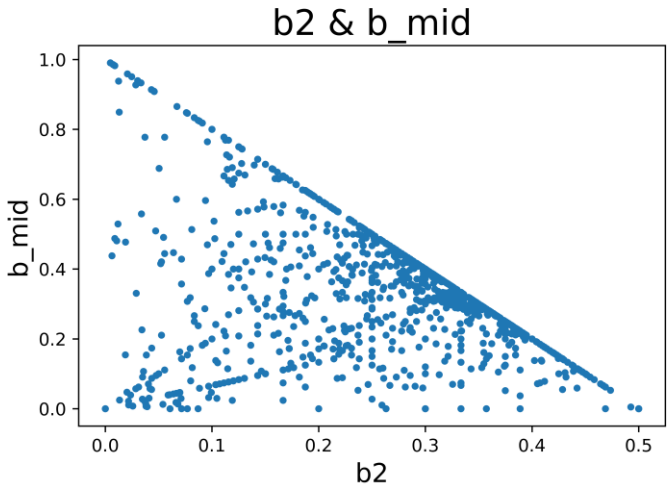


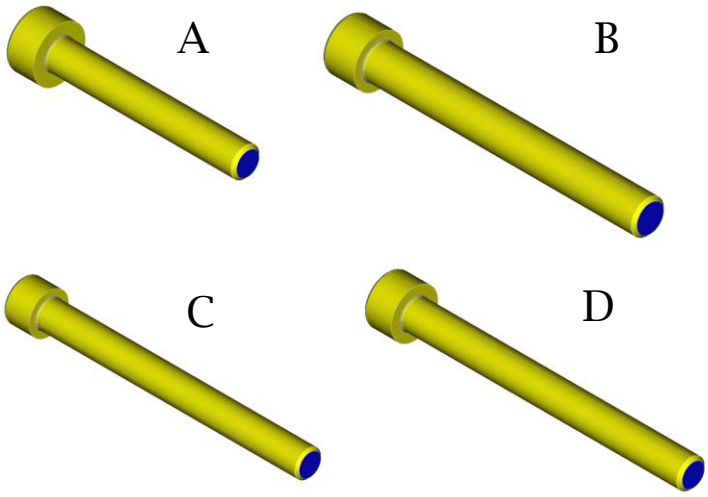
Figure 12: Linear relation between b2 and b_mid.

3.4 Results

The following Figure 13-15 show examples of geometrically similar CAD-models that were assigned to the respective clusters. The assigned data elements in each cluster are representative of the CAD-models from which corresponding information was obtained. From this context, CAD-models could thus be directly assigned to the data elements in clusters. In order to be able to present a result about the constellation of cluster-internal phenomena, some clusters were selected, whose assigned CAD-models can be viewed. The coloring of the model surfaces corresponds exactly to the information in the text-based fingerprints. The purpose of this method of representation is to simplify the interpretation of assignments since the influence of the surface types is also significant for the result. However, it should also be noted that the CAD-models may have different scaling for the purpose of clarity. Since the parameters of comparison were recorded independently of the scaling of the models, this circumstance does not play a role.

Worth mentioning is the cluster depicted in Figure 13, which exclusively contains standardized bolts. Here, only some of the CAD-models of the cluster are shown. Looking at the corresponding CAD-models, it seems that they are indeed bolts of this standard series, which are available in different dimensions. This cluster is representative of several others, which also contain exclusively standard-compliant components. The cluster contains a total of 31 elements and is therefore the cluster which contains the most CAD-models according to the mentioned database of approx. 2000 test parts. Due to the small deviation of the characteristic values of all included models, no CAD-model can be found that does not comply with the standard.

Figure 14 shows a cluster in which components can be recognized whose shape is defined mainly by rectangles. Accordingly, these surfaces are also assigned to the surface type "four" by the red coloring. An essential factor for the grouping of these elements is therefore certainly the expression zero of the characteristic ratio_round, because the CAD models have no surfaces of the surface type "round". Since these objects also have similar size dimensions, the influence of the two parameters of comparison from the bounding box group cannot be dismissed.



	ratio_round	l1	l2	b1	b_mid
A	0.758	0.823	0.098	0.827	0.068
B	0.758	0.871	0.071	0.827	0.068
C	0.758	0.852	0.081	0.827	0.068
D	0.758	0.867	0.072	0.827	0.068

Figure 13: Geometrically similar screws from one cluster.

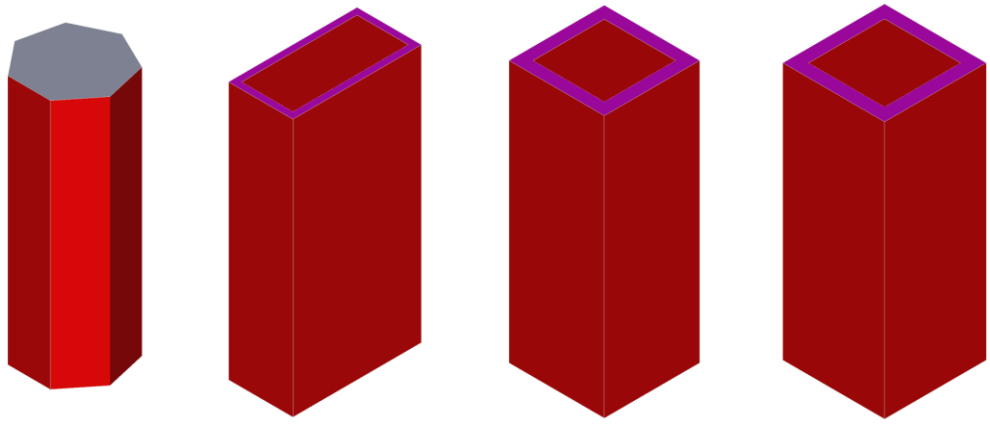


Figure 14: Geometrically similar box-shaped models from one cluster.

The last example shows one of the most geometrically complex cluster. The CAD-models shown in Figure 15 all appear to have a basic cylindrical shape and teeth evenly distributed around the circumference. It is also noticeable that the surface of most CAD-elements within the cluster has a violet color, which indicates the surface type "plane" according to the applied fingerprint method. The models also share a comparable ratio of parameters from the bounding box group, as they occupy very similar size ratios. All included CAD-models can be assigned to a gear shaped machine element.

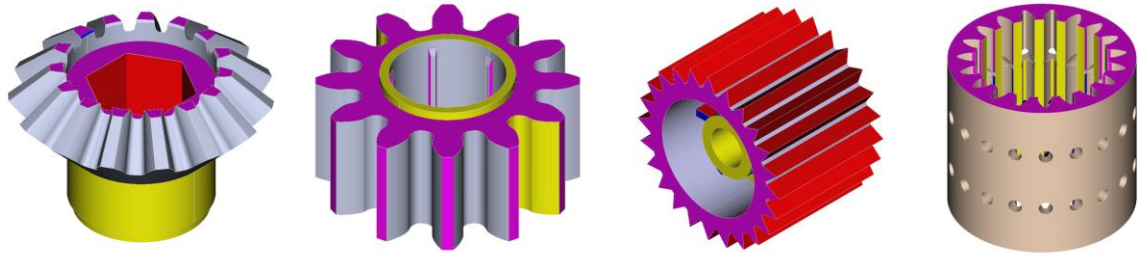


Figure 15: Geometrically similar models with gearing from one cluster.

4 DISCUSSION AND CONCLUSIONS

The application of the k-means-algorithm for the realization of an automatic classification of CAD-models was carried out on the basis of the presented fingerprint method and provided noteworthy findings. The images of exemplary CAD-models allow the realization that the data set consists of many different types of components. In the compilation of the CAD data set, the frequency of components of a particular type was not a primary consideration since controlling for this would ultimately result in only some manifestations of CAD-models being considered for the groupings. However, since the diversity of engineering designs can be assumed to be almost limitless, it is appropriate to include this fact in the compilation of the data collection as well. However, it is important that the size of the database grows with this principle.

Another interesting aspect relates to technically motivated features of a CAD model. These features include, e.g., drill holes, gradations, curves or milling pockets that are additionally added to the models. It was found that these features are also taken into account in the fingerprint and influence the result accordingly. A hole, e.g., creates new, inner surfaces which are assigned to the surface type "round" due to their appearance. Now, two identical components can be provided with different features, depending on the technical application or manufacturing method. E.g., if holes are added to one CAD-model, additional yellow surface centers are added to the fingerprint of this model. As a consequence, the expression of the parameter group "density ratio" as well as the expression of ratio_round change in correspondence to the part without corresponding features. The comparison of these two CAD-models could possibly lead to the conclusion that those parts do not belong together. Drilled plate-like parts, on the other hand, are grouped with cylindrical models because the amount of drilling raises parameter ratio_round to a level that is typical for cylindrical or annular parts. A solution to this circumstance would be to delete such features. In the section on state of the art, methods are presented that describe the suppression of features like that. If such a method is applied to the CAD-models before the fingerprint is extracted, the influence of the features can be avoided.

At the same time, however, it can also be mentioned at this point that apparently standard-compliant CAD-models were assigned to clusters in which only CAD-models of the same standard group were found. This case occurred several times, with which the statement can be made that the grouping of standard components can be carried out particularly successfully. Consequently, it can be assumed that the application of the k means algorithm to a standardized database leads to good results.

It was assumed that the resulting clusters would contain a larger set of CAD models with high optical diversity. However, tendencies were expected within the clusters that would allow approximate categorization of the cluster to a specific part shape. The actual result of the work, however, results in a larger set of clusters, in most of which there are only a few countable elements. The elements of most clusters are similar to the extent that the categorization of the

clusters to certain geometric shapes can be made. In some cases, a cluster can even be described by the name of a particular component.

The problem of classifying CAD-components is solved. The part to be classified can only be assigned to the clusters that were created with the addressed k-means model. It was mentioned that the resulting clusters still show compositions which intuitively turn out to be rather inappropriate. Some cases prove that the models are grouped strictly according to geometrical criteria, but the technical function of the parts is not taken into account, which can lead to surprising results. However, there are also cases in which the clusters concerned have elements that are very similar to each other, thus allowing an exact class assignment. In conclusion, the classification of common machine elements, such as bolts, nuts or washers, leads to good results since norm-similar objects are grouped successfully. In the case of more complicated designs, the approach to individual design of the CAD-models is more noticeable, resulting in clustering of components that make categorization of the corresponding cluster more difficult.

It was elaborated that the classification of CAD-models can be realized with the chosen approaches. It was also addressed that the size of the database plays a relevant role for the development of an appropriate ML-model. Thus, for the continuation of the present research, a k-means model based on a larger CAD data collection should be generated. Also, an approach can be incorporated in which the technical attributes of the CAD-models, such as holes, are suppressed as a precaution. These attributes are often related to the future assembly of the component and are therefore rather irrelevant for categorization. Another entry point for continuing the research is offered when optimizing or selecting the extracted parameters of comparison. The generated parameters are based on basic properties such as the center of gravity, the density or the edge dimensions, which are in principle easy to reproduce.

Robin Roj, https://orcid.org/0000-0002-7793-9791
Maxim Sommer, https://orcid.org/0000-0002-5755-0105
Hans-Bernhard Woyand, https://orcid.org/0000-0001-7677-6487
Ralf Theiß, https://orcid.org/0000-0002-4325-9413
Peter Dültgen, https://orcid.org/0000-0001-8257-9965

REFERENCES

[1] Sommer, M.: Automatisierte Klassifikation von CAD-Bauteilen, University of Wuppertal, Mechanical Engineering and Safety Engineering, Chair of Mechanical Engineering Informatics, Germany, 2020.
[2] Autodesk, Vault, https://knowledge.autodesk.com/support/vault-products/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/Vault-About/files/GUID-87D9CA09-9881-4506-9465-0677392BCD7E-htm.html, Accessed 06 January 2021.
[3] ISD Software und Systeme GmbH, HELiOS - the flexible PDM system, https://www.isdgroup.com/en/products/pdm-product-data-management/, Accessed 06 January 2021.
[4] simus systems GmbH, classmate CAD, https://www.simus-systems.com/en/classmate-cad-en/, Accessed 06 January 2021.
[5] SIMUFORM Search Solutions GmbH, Similia, https://www.simuform.com/, Accessed 06 January 2021.
[6] Siemens Aktiengesellschaft, Geolus Shape Search, https://www.plm.automation.siemens.com/global/de/products/plm-components/geolus.html, Accessed 06 January 2021.
[7] Dassault Systèmes, EXALEAD ONEPART, https://www.3ds.com/products-services/exalead/products/exalead-onepart/, Accessed 06 January 2021.

[8] Shi, Y.; Zhang, Y.; Xia, K.; Harik, R.: A Critical Review of Feature Recognition Techniques, Computer-Aided Design & Applications, 17(5), 2020, 861-899. https://doi.org/10.14733/cadaps.2020.861-899
[9] Al-wswasi, M.; Ivanov, A.: A novel and smart interactive feature recognition system for rotational parts using a STEP file, The International Journal of Advanced Manufacturing Technology, 104, 2018, 261-284. https://doi.org/10.1007/s00170-019-03849-1
[10] Mun, D.; Kim, B. C.: Extended progressive simplification of feature-based CAD models, The International Journal of Advanced Manufacturing Technology, 93, 2017, 915-932. https://doi.org/10.1007/s00170-017-0491-y
[11] Sun, L.; Zhang, B.; Li, B.; Yin, W.: CATIA V5 Robust Design Method to Prevent Feature Failure, International Conference on Automation, Mechanical Control and Computational Engineering, April 24-26, Ji'nan, China, 2015, 422-427, https://doi.org/10.2991/amcce-15.2015.81
[12] Li, G.; Long, X.; Zhou, M.: A new design method based on feature reusing of the non-standard cam structure for automotive panels stamping dies, Journal of Intelligent Manufacturing, 30, 2017, 2085-2100. https://doi.org/10.1007/s10845-017-1368-5
[13] Liu, Z.; Wang, L.: Sequencing of interacting prismatic machining features for process planning. Computers in Industry, 58, 2007, 295-303. https://doi.org/10.1016/j.compind.2006.07.003
[14] Nasution, M. K. M.: Modelling and Simulation of Search Engine. Journal of Physics Conference Series, 801(1), 2017, 012078. https://doi.org/10.1088/1742-6596/801/1/012078
[15] Hilaga, M.; Shinagawa, Y.; Kohmura, T.; Kunii, T. L.: Topology matching for fully automatic similarity estimation of 3D shapes, SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, August 12-17, Los Angeles, USA, 2001, 203-212. https://doi.org/10.1145/383259.383282
[16] Lupinetti, K.; Pernot, J.-P.; Monti, M.; Giannini, F.: Content-based CAD assembly model retrieval: Survey and future challenges, Computer-Aided Design, 113, 2019, 62-81. https://doi.org/10.1016/j.cad.2019.03.005
[17] Qin, F.-W.; Li, L.-I.; Gao, S.-M.; Yang, X.-L.; Chen, X.: (2014), A deep learning approach to the classification of 3D CAD models. Journal of Zhejiang University Science C, 15(2), 2014, 91-106. https://doi.org/10.1631/jzus.C1300185
[18] Ip, C. Y.; Regli, W. C.: Content-based classification of CAD models with supervised learning. Computer-Aided Design and Applications, 2(5), 2005, 609-617. https://doi.org/10.1080/16864360.2005.10738325
[19] Jayanti, S.; Kalyanaraman, Y.; Ramani, K.: Shape-based clustering for 3D CAD objects: A comparative study of effectiveness. Computer-Aided Design, 41(12), 2009, 999-1007. https://doi.org/10.1016/j.cad.2009.07.003
[20] Rucco, M.; Giannini, F.; Lupinetti, K.; Monti, M.: A methodology for part classification with supervised machine learning. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 33, 2019, 100-113. https://doi.org/10.1017/S0890060418000197
[21] Roj, R.: Transformation of VRML-files into graph structures in order to detect similarities and build clusters, INES 2017 - 21st IEEE International Conference on Intelligent Engineering Systems, October 20-23, Larnaca, Cyprus, 2017, 103-108. https://doi.org/10.1109/INES.2017.8118536