Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: https://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/17217

To cite this version :

Katia LUPINETTI, Marina MONTI, Franca GIANNINI, Jean-Philippe PERNOT - Content-based CAD assembly model retrieval: Survey and future challenges - Computer-Aided Design - Vol. 113, p.62-81 - 2019

Any correspondence concerning this service should be sent to the repository Administrator : scienceouverte@ensam.eu

Content-based CAD assembly model retrieval: Survey and future challenges[☆,☆☆]

Katia Lupinetti [a,b,*], Jean-Philippe Pernot [b], Marina Monti [a], Franca Giannini [a]

[a] Istituto di Matematica Applicata e Tecnologie Informatiche "Enrico Magenes", CNR Via De Marini 6, 16149 Genova, Italy
[b] Arts et Métiers, LISPEN EA 7515, HeSam, Aix-en-Provence, France

ARTICLE INFO

Keywords:
Assembly retrieval
Assembly similarity evaluation
Assembly matching
Knowledge representation
Knowledge extraction

ABSTRACT

Currently, the content-based retrieval is a problem of major interest in several different fields and, focusing on mechanical engineering, many approaches exist to compare and retrieve single CAD parts, to evaluate shape similarity, to extract features and to segment models. However, most of the proposed approaches do not take into account all the key characteristics of an assembly model, such as the relationships between its components, and the different levels according to which two assembly models can be considered similar, i.e. either globally, partially, or locally. For these reasons, the retrieval of CAD assembly models still faces challenges to fully satisfy designers' expectations. The aim of this paper is to review the state-of-the-art of works addressing the CAD assembly model retrieval and to identify future challenges and possible research directions. Firstly, the paper highlights the user requirements for CAD assembly model retrieval and proposes a set of criteria for analyzing the available methods grouped into the following macro-categories: objective, assembly characterization, assembly descriptor, query specification and type of similarity. Secondly, it describes and characterizes the available methods by organizing them according to the adopted criteria. Finally, it discusses the open issues and future challenges.

1. Introduction

Designing and developing a product is a complex cyclic and iterative process, which includes the specification of the various constituting functional sets and related composing parts. Each person taking part in the Product Development Process (PDP) makes use of specific knowledge needed to define functional specifications, mapping from function requirements to physical description, feasibility and usability [1]. It follows that knowledge in product design has a wide range of meaning and its representations depend on the context. To stay competitive on the market, companies have to capitalize, transfer and communicate knowledge within their teams [2,3].

Today, managing efficiently the knowledge associated with the product, handling a possible huge amount of heterogeneous digital data located on different sites and supports, being more dynamic in the decision making, being more reactive and flexible to the evolutions of the market has become a clear differentiation criterion. This is at the base of the fourth industrial revolution, commonly known as Industry 4.0 [4].

Currently, most of the semantic knowledge associated with the multiple representations at multiple resolutions of a product is managed by PDM (Product Data Management) and PLM (Product Life-Cycle Management) systems, which handle much more information than only the geometric data. Ideally, all the digital data associated with a product during the PDP should be stored in a well-structured manner within those systems. Depending on the companies, there exists a large variability regarding the amount and quality of available digital data. This is not only true for the data themselves but also for the metadata which can be attached by means of attributes. This variability or even lack of data documentation makes difficult the access and reuse of the relevant data and knowledge. Actually, both explicit and implicit information can be necessary when searching digital data. The first set corresponds to all the data which are directly available from the database, whereas the second may require more sophisticated reasoning processes to extract and interpret the meaningful data [3]. Thus, even PDM and PLM systems alone do not fully meet the Industry 4.0 requirements of an autonomous and efficient knowledge exchange and retrieval [4].

☆ This paper has been recommended for acceptance by S Hahmann.
☆☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.cad.2019.03.005.
* Corresponding author at: Istituto di Matematica Applicata e Tecnologie Informatiche "Enrico Magenes", CNR Via De Marini 6, 16149 Genova, Italy.
E-mail address: katia.lupinetti@ge.imati.cnr.it (K. Lupinetti).

Independently of the use of PDM and PLM systems for the product data organization, the 3D CAD models are generally considered as central representations used to convey the knowledge and information along the PDP. Therefore, they provide suitable keys to access the digital data, and thus knowledge, related to the products. As a consequence, being able to evaluate similarity between 3D CAD models and retrieve the corresponding digital data within the Digital Mock-Up (DMU) has become mainstream in the context of Industry 4.0. Several issues are to be considered when developing such similarity evaluation approaches: (i) the concept of similarity strongly depends on the application context and objectives. Indeed, the similarity evaluation relies on the type of knowledge that the user wants to gain, and this influences what can be considered similar. Those issues are even more challenging when considering assemblies of CAD models; (ii) to get a meaningful similarity evaluation and to filter the results, extrinsic information is not enough, thus it is necessary to extract and use intrinsic information; (iii) the size of the databases has grown up exponentially in the last few years and a DMU can incorporate more than 1 million parts representing several terabytes of data [5], thus it is increasingly challenging to handle a large amount of produced data and to develop efficient searching and browsing methods and tools [6].

There exist many methods for content-based parts retrieval dealing with models represented as both 3D meshes and B-Rep [7,8]. They can be grouped according to the different approaches, e.g. shape-based [9–15], feature-based [16,17] or topology-based [18–21]. Some of them can also detect partial similarities, i.e. models that are similar only for a subset of their shape [22–25].

Although these techniques are able to retrieve single parts of assembly models, they do not take into account all the diverse aspects characterizing an assembly such as the relationships between its parts, and thus they are limited for assembly retrieval. Actually, CAD assembly models are designed to perform specific kinematic functions that cannot be detected without analyzing how the single parts interact [26,27]. Moreover, an additional issue derives from the plurality of the similarity levels according to which two assembly models can be considered similar. Indeed, two assemblies may be *globally* similar, but also *partially* similar; where partial similarity may be further split into *part-in-whole* (i.e. input model completely contained in a retrieved model) and *whole-to-whole by partial matching* (i.e. a subpart of the input model is similar to a subpart of the retrieved model). In the following, the first is referred as *partial similarity* and the second is indicated as *local* similarity. In the example of Fig. 1, models M1 and M2 are globally similar, they are also partially similar to M3 and M4 as the first two are contained in the last two ones; finally M3 and M4 are locally similar since they share similar subparts.

More recently, efforts have been devoted to exploit the identification of some meaningful sub-parts of objects for the model classification and retrieval in selected contexts. For instance, in computer graphics and computer vision, recent works have investigated the use of deep learning techniques to evaluate shape similarities [28–31]. In general, these methods are not yet effective for the retrieval of CAD assembly models, because they evaluate shape similarity neglecting other important features characterizing the design of a product. Therefore, in this context, the criteria used to assess the similarity cannot fully capture all the knowledge involved in the retrieval of CAD assembly models. For instance, to recognize local similar features, the method proposed by Qi et al. [31] performs a segmentation that does not consider at all the design intent, as well as the more general information embedded in a DMU. Their reasoning, is limited to the geometric information available from the mesh representations. Even other segmentation strategies as the one proposed by Huang et al. [32], which aims at segmenting objects by identifying the possible joints, are not adequate in the mechanical engineering field since they are simple view-based approaches that do not handle information about the technological solutions adopted for the definition of the joints. To clarify this difference, Fig. 2 shows an example of two different chairs. The legs of the chair in Fig. 2(a) are welded together creating a support structure that is screwed with the seat; while the legs of the chair in Fig. 2(b) are represented as single parts in the CAD model and they are linked by dovetail joints. In this example, a traditional segmentation process splits the objects regardless of the building technology. In other words, it considers only how an object looks like while neglecting other important information, such as the kinematic links. Finally, even if adopting a deep learning approach in the matching process seems promising when compared to traditional methods based on graph matching, the lack of proper datasets of CAD assembly models makes such implementation difficult (see Section 6).

Similarly, methods for the comparison of 3D scenes, as the one proposed by Paraboschi et al. [33], are suitable to recognize global as well as local object similarities at the level of the shapes, but they fail to identify internal mechanisms which typically characterize certain products. Indeed, for instance, there exist a huge amount of mechanical systems made of gears, and what characterizes these products is the arrangement of the gears, which typically has an influence on their mechanical characteristics (e.g. different gear reduction rate and transmission yield).

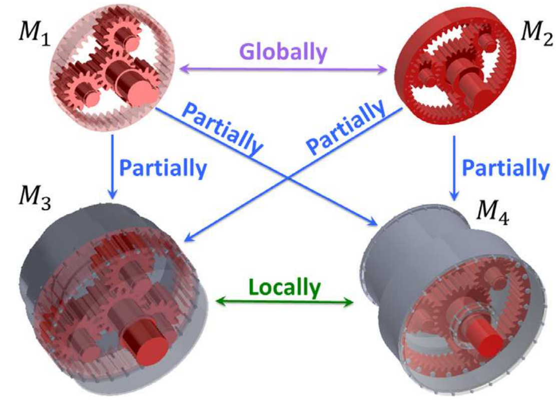


Fig. 1. Different types of similarity among assembly models: local, partial and global similarity.

(a) Chair with legs welded to a common structure

(b) Chair with legs linked by dovetail joints

Fig. 2. Example of objects with similar shape but made of different components and types of joints.

This paper reviews techniques addressing the evaluation of similarities between CAD assembly models, focusing on the information implicitly embedded in the CAD models. The contribution is threefold: (i) definition of a set of criteria for the comparison and categorization of the existing CAD assembly retrieval techniques; (ii) an in-depth analysis and a systematic characterization of the existing techniques with respect to the identified criteria; (iii) an exploration of the current issues and future challenges. The paper is organized as follows. Section 2 provides an overview of the content of CAD assembly models introducing the adopted terminology and highlighting the issues characterizing CAD assembly model representations. Section 3 describes how the application context may influence the similarity evaluation. The criteria adopted for the analysis of the retrieval methods are introduced in Section 4, while Section 5 contains the systematic analysis of the assembly retrieval techniques. Finally, Section 6 discusses the current limitations and future challenges regarding the retrieval of content-based CAD assembly models.

2. Background: elements of a DMU relevant for assembly retrieval

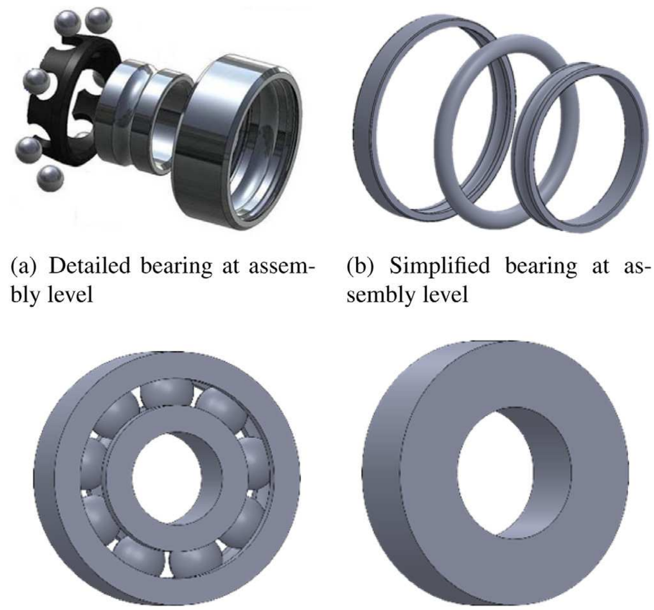
DMU represents a clearly defined set of data in the product model, whereas the term "product model" includes all of the information gathered during the PDP [34]. Generally, a DMU consists of three types of data [35]: *geometric data* (i.e. geometric description of the parts involved in an assembly model), *product structure* (i.e. how the parts are gathered together) and *attributes* (i.e. metadata referred to parts or to their relationships). Despite this commonly adopted decomposition structure, several alternative implementations might be adopted in existing CAD systems, thus complicating the design of efficient retrieval techniques. For instance, the positioning of the parts can be performed in different ways, and the DMU can be more or less simplified. These aspects are discussed in the next sections.

2.1. Geometric data

Geometric data describe the shape of components (parts or subassemblies) and are generated by CAD modelers. To represent a solid object, the boundary representation (B-Rep) is the defacto standard in commercial CAD systems. Elements used in a B-Rep are shells, faces, loops, edges and vertices, as well as the corresponding geometric information, e.g. surface types and parameters, curve equations and point coordinates. In addition, a B-Rep describes how the elements are related to each other, i.e. the topology. Moreover, the history of construction sometimes is also represented as a building tree, i.e. it stores the order of the features used to design a part. Anyhow, this tree is not unique, because parts can be built in different ways and different features can be associated with them. Depending on the steps of the PDP, the building trees may however not be available.

The geometric description of the B-Rep elements can use analytic or parametric representations. Here, particular attention has to be paid to the vocabulary adopted in the literature. Sometimes authors refer to CAD models even though the proposed approaches deal with meshes obtained by tessellating CAD models, which is significantly different. Similarly, some authors refer to assembly models even though they manipulate collections of meshes [29,30,36,37]. This state-of-the-art focuses on the methods which make use of B-Rep CAD models defined by analytic and parametric representations, being those adopted in the mechanical engineering context.

Furthermore, the geometric description of the CAD models can be defined at different levels of detail depending on the lifecycle stage and the PDP organization. For instance, in the early



(a) Detailed bearing at assembly level

(b) Simplified bearing at assembly level

(c) Detailed bearing at part level

(d) Simplified bearing at part level

Fig. 3. Example of multiple representations and multiple resolutions related to a bearing component.

design stage, a CAD model is usually roughly detailed in all its components; later for simulation activities some components are completely detailed and others (considered less important with respect to the simulation objectives) can be simply drafted or even removed; at the final stage all the components to be manufactured have to be completely specified. Similarly, some parts may not be completely detailed because designed and produced by an external company. This refers to the notion of multiple resolutions of the CAD models, which has to be taken into account when developing retrieval system.

In addition, standard components (e.g. screws, nuts, bearings, gears, seals or circlips) are often imported from supplier catalogs and/or 3D databases. Therefore, they are not necessarily designed using the same CAD modeler, and also the modeling strategy may differ. Thus, for a given component, depending on the supplier, multiple geometric representations and multiple resolutions may exist. Fig. 3 shows an example where a bearing is designed in four different manners: two as assembly models and two as parts. Moreover, the representation can be complete allowing to recognize the bearing (as in Fig. 3(a) and Fig. 3(c)), or simplified with some idealized shapes (as in Figs. 3(b) and 3(d)) which will be hardly identified as a bearing by a traditional retrieval approach.

Finally, the possibility to represent a product in many different ways prevents the use of the number of elements of two assemblies as an effective similarity indicator and, more generally, this large multiplicity may affect the capacity to retrieve models in a completely automatic way. Indeed, when the shape is idealized (as for the bearing in Fig. 3(d)) it is hard understanding what the part corresponds to. For instance, it could represent a simplified bearing, a simplified gear, or a simplified seal. Sometimes, exploiting the information on the surrounding context of the part can help to retrieve the correct interpretation.

2.2. Product structure

Designing an assembly model is a complex process aiming at creating a product satisfying predefined requirements by a

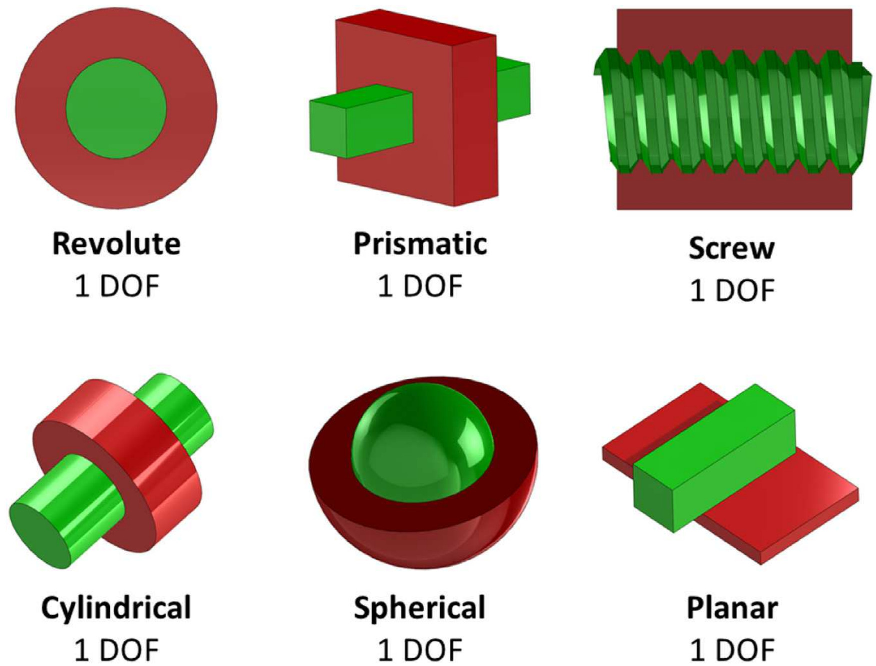


Fig. 4. Lower kinematic pairs.

combination of components accomplishing specific functions. The components of an assembly model may be gathered together using a hierarchical and logical structure of dependence among the designed parts. Such product structuring is not unique and is used to organize product data in a manner appropriate to the designers' needs.

The most commonly adopted product structures are the as-designed (i.e. aggregating parts by their function, such that each subassembly represents a unit that performs a specific function) or as-planned (i.e. reflecting how parts have to be manufactured or assembled from a manufacturing or a process planning perspective) structures [38]. Depending on the companies and stages of the PDP, other structures can be adopted, such as maintenance or quality structures. Sometimes, parts can be organized with respect to their relative positioning for visualization purposes. They can also be grouped according to their material to speed up the preparation of advanced simulations.

Finally, the product structure is usually stored separately from the geometry, even if modern CAD systems allow including it in the CAD models. When, the product structure is not available, all the parts are gathered together under a unique root node. Unfortunately, this variability in the way assembly models can be decomposed and structured is not always taken into account by the methods in literature.

2.3. Attributes

Besides component geometry and product structure, annotations are used to express explicitly some geometric properties such as major/minor diameters, pitch, or number of threads [39]. Since a DMU can be simplified and details may not be fully defined, additional attributes can be used to further characterize parts. For instance, component material, and physical properties are represented as annotations. They are necessary to enable the manufacturing of a product [40] or to perform simulations. In the end, other attributes are used to identify name, number and version of a product, to distinguish its status and maturity level in the PDP and to provide details about description, material and product manufacturing information.

The above mentioned explicit information may be present in the DMU as attributes, but it is not mandatory. Moreover, the absence of conventions among designers and the variability against the industrial context make challenging to exploit this information in retrieval systems. Actually, this type of information is not robust and of little use for CAD retrieval [7]. Thus, in this review, the retrieval techniques which try to make use of such unreliable information have been identified.

2.4. Components' positioning

In addition to the definition of the product structure, designing an assembly model requires localizing each part in the 3D space [41]. When considering physical objects, components are positioned relatively to the others by means of contacts. Similarly, in the DMU, parts are positioned to characterize the possible relative displacements. However, this information can be not always available or designers can simply use homogeneous transformations to position parts. In addition, as discussed in Section 2.1, the DMU does not always perfectly represent the real configurations and some shapes may be simplified. Hence, the pure geometric information stored in the DMU to assess the contacts between two components can be ambiguous. To circumvent this limitation, designers often make use of extra-information to explicitly encode and constrain the relative positions of the parts. The specification of the contacts is then performed through at least one of the following solutions [39]:

- **Kinematic links (or joints)** characterize the relationships between parts. They determine the positions of the components as well as the allowable movements, i.e. the allowed degrees of freedom (DOF). The kinematic links are divided into two groups: lower kinematic pairs and upper kinematic pairs. A kinematic pair is said to be a lower pair if the involved parts have surface area contact between them. Different lower kinematic pairs can be identified according to the types of surfaces involved in the contact. The possible lower kinematic pairs are depicted in Fig. 4. Interestingly, these kinematic pairs are not necessarily linked to the shapes of the involved surfaces, e.g. the kinematic pair

Fig. 5. Example of geometric constraints.

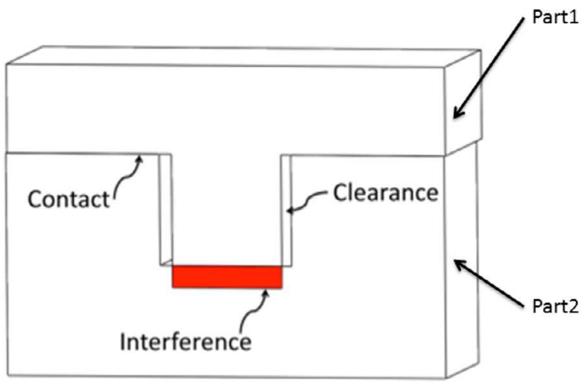


Fig. 6. Possibles interfaces between parts of an assembly model.

of two cylindrical surfaces can be a screw even though the threads have not been modeled geometrically on both parts. An upper kinematic pair arises when two surfaces are constrained to remain in contact along a common line or at a common point [42]. Ball bearings have this type of kinematic pair, as the contacts between the balls and the inner and outer rings are punctual.

- **Assembly constraints** determine the relative position of geometric entities (i.e. faces, edges or vertices) of parts of an assembly model. Typically, geometric constraints between parts include: parallel, perpendicular, coincident, tangent, concentric, distance and angle (Fig. 5). Kinematic pairs can be defined through a bundle of assembly constraints, but assembly constraints can also be used alone, without any definition of kinematic pairs.
- **Absolute positions** when parts are placed in a single 3D reference frame using homogeneous transformations to define the affine transformation matrix for each object.

Today's CAD systems provide capabilities to easily specify and store the positions as well as the possible relationships between parts. However, when considering large DMUs made of several hundreds of parts, storing, updating and modifying those relationships can rapidly become very difficult, even impossible. Thus, the parts in a DMU are often simply gathered in hierarchies of subassemblies and only the absolute positions are stored, i.e. without information about what parts are connected and how. In this literature review, particular attention is paid to the retrieval methods that assume the availability of this information.

2.5. Interfaces modeling

Unrealistic or unrealizable configurations may be present in an assembly model [39]. This is the case, for instance, of volumetric interferences (i.e. self-intersections) between parts of an assembly model [43,44]. In an assembly, interfaces may be grouped into interferences, contacts and clearances as shown in Fig. 6.

In real-life situations, some of these configurations are not possible. In the DMU, they are generated by some mistakes or designed on purpose to convey a certain meaning [35]. An interference is a non-realistic configuration since it implies overlapping volumes of two components in a product, which is not possible for physical objects. Nevertheless, unrealistic interferences may be created on purposes, such as the intersections among screw and nut threads, or when considering flexible parts, like springs, seals and insulating parts, or when designing parts assembled by shrink-fitting. Thus, some of these configurations can be interpreted as imprecise positioning, while others are deliberate artifacts reflecting some conventional meanings [39]. Clearly, the existence of such ambiguous configurations may affect the similarity evaluation process, and two models, corresponding to similar physical objects, may be recognized as dissimilar simply because they are modeled differently. Thus, in this state-of-the-art, it is important to distinguish the retrieval methods which are able to deal, or not, with these unrealistic configurations.

2.6. Conclusions

As discussed in the previous sections, even though engineers can spend a lot of time designing and enriching industrial DMUs, there often exists a gap between the generated assembly models and the corresponding real-life physical products. Unrealistic interfaces, simplified shapes, ambiguous configurations, missing information, large variability in the way CAD models can be designed and assembled, inconsistency and unreliability of the available datasets are issues that can affect the effectiveness of a retrieval system. Vilmart et al. also reached the same conclusion and emphasized the importance to have an assembly description that is independent of any user intervention [45]. To this aim, the intrinsic properties of an assembly model (e.g. parts occurrences, symmetries, patterns, mating information) should be used to characterize the description of CAD assembly models. In the sight of these considerations, this review notably aims to understand how the existing retrieval approaches describe an assembly model and if the used information is provided manually by the user, or if it is automatically extracted.

3. Application scenarios and similarity criteria

Assembly retrieval may be of interest in several activities of the PDP where the criteria for evaluating similarities may obviously change according to the objective. To underline the importance of evaluating assembly similarity according to different points of view, this section provides an overview of some application scenarios that can benefit from assembly model retrieval, highlighting for each of them the most appropriate type of similarity and the criteria according to which it may be meaningful for the user to evaluate similarity [46,47].

3.1. 3D model reuse

In the design of new products, it is common practice to reuse existing 3D models to include components previously designed using them as originally designed, or making slight changes to meet new requirements [47–49]. To avoid starting the detailed design phase from scratch and to capitalize on previous knowledge, engineers might be interested in examining any existing solution considered similar to their needs. These solutions can include also components provided by third parties thus not fully detailed. To this end, it may be useful to start a rough design of the new component and use it to search the similar components over the whole dataset. This recovery process should identify a restricted collection of assemblies that contain parts with similar shapes and comparable assembly conditions. In this case, it is useful to start with a rough query, i.e. a query in which the shape of the components is not fully detailed but just sketched. Thus, the similarity of the part shape must be evaluated at a level of detail that does not consider, for example, minor characteristics such as fillets and chamfers. Here, the retrieved objects may be similar to the query either partially, locally or globally. Sometimes, it may be useful to reuse and update previous designs when it is necessary to replace a certain type of product components. For example, when due to the working conditions, a type of bearing adopted in an assembly model needs to be replaced with another type capable of supporting more load. Retrieving all the models containing that specific component assembled in similar conditions allows identifying the products, which may benefit of the component substitution. In addition, the identification of the position of these defective components in an assembly helps to evaluate procedures and costs necessary for their replacement and to correctly update the related CAD models. Here, the type of interesting similarity is partial or local and the criteria for evaluating the similarity can be stricter, involving, for instance, also the dimensions of the components and the number of constituent parts together with all the information on shape and mutual relationships.

3.2. Product information reuse

This application scenario corresponds to the process of mining a database to retrieve design information and documentation associated with a given product. Generally, it allows obtaining useful knowledge for the design of a new product: technical information, production processes and costs associated with similar products previously developed [50]. For instance, if a designer wants to retrieve assembly instructions of some products, it might be useful to retrieve models considered either globally, partially or locally similar according to the mating conditions between components.

3.3. Product standardization and rationalization

Standardization is the process of defining common characteristics among a set of components so that they are compatible with each other. This process allows the rationalization of products by eliminating very similar components or by outsourcing products and product variations, thus reducing the size of the product portfolio to be developed.

These practices allow a considerable time saving, especially in case of complex devices with many parts which may require a complex design and/or production process. Here, the evaluation of the similarity is mainly local and can involve the functionality of a component as well as some information on how it is linked to other components. For instance, if a company aims to standardize the steering wheel of certain cars, then the specific shape of the steering wheel is not relevant, while the most important characteristic is how the wheel is linked to the drive shaft by means of external mating surfaces.

3.4. Maintenance planning

Maintenance refers to those activities necessary to preserve the status of a product preventing its damage due to the aging and deterioration of components. To this aim, the retrieval of similar assemblies meant as the identification of similar components in a set of assembly models, and the knowledge of their rate of wear and tear helps to optimize the management of the stocks in the warehouse. Also in this scenario, the user aims to retrieve a specific component included in a set of assembly models (thus locally or partially similar assemblies with strict similarity criteria involving shape and mutual relations).

3.5. Reverse engineering

In mechanical-field, reverse engineering is the process that creates a 3D digital model starting from a physical object. The reconstruction of a digital model starts capturing data from real objects, where the acquisition may be done through different devices, as a camera, a laser scanner or a 3D computed tomography (CT). Based on the generated point clouds, designers often have to follow a tedious and time-consuming patch-by-patch reconstruction strategy to come up with a fully reverse engineered B-Rep CAD model. This is even truer when considering the reverse engineering of CAD assembly models. Thus, being able to shift from a patch-by-patch to a part-by-part modeling strategy can drastically speed up the CAD assembly reconstruction process. Here, the ability to retrieve models into a database can facilitate some reconstruction operations, and in this case, the type of query may rely on the adopted acquisition tool. If a laser scanner has been used, then the aim is to retrieve models with a similar outer shape, while a 3D CT allows the possibility to investigate also the similarity between part relationships. In this scenario, all the three types of similarity may be interesting.

4. Adopted criteria to review the literature

In this section, criteria to analyze the existing methods that address the identification of similarities among assembly models are identified and described. The presented criteria are grouped into the following five macro-categories: objective, assembly characterization, assembly descriptor, query specification and type of similarity. Those categories are used to set up Table 1, which gathers together the synthetic evaluation of the reviewed methods with respect to the considered comparison criteria.

4.1. Objective

The process of retrieving assembly models benefits various stages of the PDP, where a different retrieval purpose may identify different similarity requests. For this reason, this criterion is used to highlight the specific objectives addressed by the different methods.

4.2. Assembly characterization

To understand according to which criteria the similarity can be assessed by the works present in the current state-of-the-art, it is important to know the type of information used to describe the assembly models in the retrieval system. Thus, the set of criteria used for the characterization of the assembly refers both to the type of data and to the knowledge that the authors use to typify the assembly model and the way the information is obtained. More precisely, it includes the so-called **Part information** criterion which expresses the geometric characteristics used to describe the assembly models components, the **Topological information** criterion which captures the type of information used to characterize the relationships between the parts, the **Annotations** criterion which states if the retrieval methods make use of such extrinsic information, and the **Functional classification** criterion to explain if the assembly descriptor elaborates somehow these data to get a functional classification of the components.

Table 1
Summary of the assembly retrieval methods.

Article	Objective	Assembly characterization									Assembly descriptor					Query specification		Type of similarity		
		Part information			Topological information						Level of components		Level of descriptor							
Title	Description	Geometric information	Shape descriptors	Statistical information	Structure	Kinematic link	Geometric constraints	Part arrangement	Annotations	Functional classification	Assembly	Part	Global	Local	Scale sensitivity	Type of query model	Completeness	Global	Partial	Local
Renu et al. [51]	Search for models with similar shapes for assembly instructions reuse	–	SD	–	–	–	–	–	–	–	–	✓	✓	-	✓	Part	●	✓	–	–
Katayama et al. [52,53]	Search for models with similar shapes, same number of components and same layout	–	2DP	–	–	–	–	–	–	–	✓	–	✓	–	✓	Assembly	●	✓	–	–
Wang et al. [54]	Search for assembly models with similar shapes	–	SD	–	–	–	–	–	–	–	–	✓	–	✓	–	Assembly	◐	✓	✓	–
Zhang et al. [55]	Search for assembly models with similar shapes	–	SD	–	–	–	–	–	–	–	–	✓	–	✓	–	Assembly	◐	✓	✓	–
Hu et al. [56]	Search for assembly models with similar shapes and composition	–	LFD	PS	–	–	–	–	–	–	–	✓	✓	–	✓	List of parts	◐	✓	✓	–
Tao et al. [57]	Search for assembly models with similar shapes and component connections for assembly plans generation	SI	–	EN, LN	–	A	–	–	✓	–	✓	✓	✓	✓	✓	Assembly	●	✓	–	–
Miura et al. [58]	Search for assembly models with similar shapes and component connections	–	AD	–	–	-	A	–	–	–	✓	✓	✓	✓	✓	Assembly	●	✓	–	–
Han et al. [59]	Search for assembly models with similar parts, constraints and function information	–	–	–	–	PC	PC	–	✓	AR	✓	✓	✓	✓	–	Text	◐	✓	✓	–
Li et al. [60]	Search for similar assembly models of injection mold design of automotive interiors	–	SD	–	A	A	–	–	✓	AI	✓	✓	✓	✓	✓	Part or Assembly	● ●	✓	✓	–
Deshmukh et al. [47,61,62]	Search for similar assembly models according to multiple assembly characteristics	C,SI	–	–	–	A	A	–	✓	US	✓	✓	✓	✓	✓	Matinggraph	◐	✓	✓	–
Chen et al. [48]	Search for similar assembly models according to multiple assembly characteristics	–	SD	–	A	PC	PC	C	–	US	✓	✓	✓	✓	✓	Assembly	◐	✓	✓	–
Zhang et al. [63]	Search for similar assembly models according to multiple assembly characteristics	C,DA,SI	–	–	A	A	–	–	–	–	✓	✓	✓	✓	–	Assemblyset	●	✓	✓	✓
Wang et al. [64]	Search for similar assembly models according to multiple assembly characteristics	–	SD	–	–	A	–	–	–	–	✓	✓	–	✓	✓	Assemblyset	●	✓	✓	✓
Lupinetti et al. [65]	Search for similar assembly models according to multiple assembly characteristics	–	3DSH	–	A	C	–	C	–	AI	✓	✓	✓	✓	✓	Assembly orGraph	◐ ◐	✓	✓	✓

4.2.1. Part information

This criterion indicates which information is used to characterize the parts in an assembly model. To guarantee a comprehensive and structured classification, it is divided into: (1) Geometric information, (2) Shape descriptors, (3) Statistical information. Then, for each category only the information used by the reviewed retrieval methods are shortlisted and explained.

Geometric information

Geometric information specifies if an approach characterizes the parts of the assembly models by using data that can be easily computed processing their B-Rep models. The types of geometric information used by the methods included in this survey are described below:

- *Curvature* (C): The normal curvature at a point P on a surface varies around the normal direction of the surface. The maximum and the minimum values of the normal curvature are named as principal curvatures and the difference of their signs can characterize the point on a surface. In this state-of-the-art, the works, which use normal curvature to characterize the shape of the parts, sample points on the faces and then evaluate the average of the different types of point.
- *Dihedral angle* (DA): A dihedral angle is the internal angle defined by two adjacent faces on an edge. According to the normals of the faces and the direction of the edge, a dihedral angle can be concave, convex or smooth.
- *Surface information* (SI): It refers to the *type of surface* underlying the faces of the parts, i.e. if a face is planar, cylindrical, conical, spherical, toroidal or other; and the *surface convexity*, i.e. convex, concave or planar.

When analyzing the reviewed approaches, the type of geometric information used is described. Methods that exploit multiple types of geometric information report multiple labels in Table 1 under the column *geometric information*.

Shape descriptors

The shape descriptors indicate how the methods characterize the shape of the parts. Shape descriptors may be computed directly from the B-Rep of the parts or they can require a preprocess to obtain a polygonal representation. This survey focuses on the type of descriptor used and not on how it has been computed. Then, for each analyzed method, only the characteristics of the adopted shape descriptors are described. The shape descriptors used by the analyzed methods are described below:

- *Shape distribution* (SD): This descriptor is used to evaluate the shape similarity of two parts. It is described by Osada et al. [10], where the 3D shape of each part is characterized by the distances of randomly sampled points on the surface of the parts. Several distances can be used to compute the shape distribution and most of the time, in the considered methods, the Euclidean distance is employed.
- *Set of 2D projections* (2DP): It is a view-based descriptor to characterize components of assembly models according to their shape regardless of their structure. Since view-based methods are not robust to translation and rotation in the space, a set of projections is collected.
- *Light Field Descriptor* (LFD): It is a view-based descriptor that collects different 2D views of a 3D model [66]. It is based on the idea that if two 3D models are similar, they also look similar from any view angle.
- *Angle distance* (AD): The angle distance is a two-dimensional distribution proposed by Ohbuchi et al. [67] for the shape retrieval of components, where the first dimension indicates the normalized distribution of distances between sampled points on a part, while the second dimension refers to the normalized distribution of inner products between the surface normal vectors.
- *Spherical harmonics* (3DSH): The evaluation of shape similarities can rely on the computation of spherical functions following the method proposed by Kazhdan et al. [68]. Here, a function (that represents an approximation of the object to be described) is decomposed into harmonics, the harmonics are then summed with respect to their frequency and the norm of each frequency component is finally computed. It results in a normalized histogram, which reports the values of the sums of the harmonics for the given frequencies. In particular, in their work, there are 544 bins in the histogram.

Statistical information

Statistical information indicates which numerical data are used to describe the parts of an assembly. Innumerable data can be used for this purpose, and the statistics used in the analyzed methods are listed below:

- *Part statistics* (PS): Two assembly models can be compared according to the parts that compose them. The works that adopt this description refer to the number of parts and how many times each part appears in an assembly model.
- Edge number (EN): This criterion indicates if the methods take into account the number of edges or of outer edges to characterize the parts of an assembly model.
- *Loop number* (LN): This criterion indicates if the methods make use of the number of loops to characterize the parts of an assembly model.

When analyzing the reviewed approaches, the type of statistical information used is described. Methods that exploit multiple types of statistical information report multiple labels in Table 1 under the column *statistical information*.

4.2.2. Topological information

This criterion helps to describe the type of information used to characterize the relationships among the parts of an assembly model. Here also, a method can make use of several of those descriptors and this is highlighted by multiple labels in the column *topological information* of Table 1:

- *Structure*: If a method makes use of this descriptor, it means that it exploits the hierarchical decomposition of the assembly models, as described in Section 2.2.
- *Kinematic links*: This descriptor refers to the relationships defined by the contacts between parts as described in Section 2.4. In the analyzed works, this information sometimes is referred as mating conditions or joints between two parts.
- *Geometric constraints*: This descriptor refers to the exploitation of specific constraints between geometric entities of two parts of an assembly model, as described in Section 2.4.
- *Part arrangement*: Since different parts can be positioned in an assembly model in several ways, this descriptor characterizes if the analyzed works are able to recognize particular part arrangements in the 3D space, e.g. repetition of some parts.

In this paper, each method is analyzed to identify whether it uses or not (−) the above-mentioned topological information. Table 1 also details the way the information is collected: (A) characterizes methods which assume that the relationships between the components of an assembly are explicitly encoded and available from the native CAD models; (C) indicates that they are automatically derived and computed from the assembly geometry; (PC) if they are partially computed (i.e. if some configurations are automatically extracted and others are manually specified by the user).

4.2.3. Annotations

As described in Section 2.3, attributes are sometimes added and attached to the assembly models to specify details on the shape of the parts or to further specify useful data (e.g. material, welding types and tolerances). Here, this criterion simply captures whether the reviewed methods use (✓), or not (−), annotations as structured metadata (e.g. ontology or thesaurus) or simple text annotations. As discussed earlier, exploiting such unreliable information cannot be considered as an effective practice for retrieval applications since it is user- and system-dependent.

4.2.4. Functional classification

Today, there is a growing interest in developing knowledge-based and semantic-based assembly retrieval systems to access and to exploit the functional characteristics of the product components and subassemblies [3,45]. Functional information can be extracted automatically by reasoning on some information available in the CAD models, or exploiting support systems that encode product data, or using user-specified data. To characterize the methods according to the techniques used to populate the functional information, this criterion can be associated with different labels: user specification (US), algorithm reasoning procedures (AR), artificial intelligent techniques (AI), or use of ad-hoc tools such as PDM/PLM systems or ontologies (ST).

4.3. Assembly descriptor

This criterion aims to characterize the assembly model descriptor adopted by the different reviewed works. In particular, it specifies the **Level of components** that are described, the **Level of descriptor** and if the assembly descriptor is **Scale sensitive**.

4.3.1. Level of components

This criterion indicates if, in the reviewed works, the assembly models are described at the level of the *Assembly*, at the level of the *Part*, or at the level of the *Feature*. At the assembly level, an assembly model is described by its parts and their relationships. At the part level, an assembly is described only through the list of its parts, and at the feature level, shape portions having specific assembly meaning are used to characterize an assembly. Since none of the analyzed methods uses the feature level description, in Table 1 this level does not appear.

4.3.2. Level of descriptor

It indicates if the assembly descriptor is able to capture local characteristics of an assembly model or if it describes the assembly under a global point of view. Thus, this criterion may take two values: *Global* or *Local*. Note that even if the descriptor is able to characterize the assembly model at the local level, this does not automatically imply that the retrieval method exploits this ability to assess local or partial similarities.

4.3.3. Scale sensitivity

This criterion specifies if the method is able (✓), or not (−), to distinguish assemblies with the same number of parts and assembled in the same order, but having different sizes. Usually, this ability relies on the type of data used to characterize the assembly models. For instance, shape distribution descriptors that compute several distances on the surface of a part are influenced by the dimension of the part if these distances are not normalized.

4.4. Query specification

A database can be queried in multiple ways according to both the available data and the user's search purpose. For instance, to retrieve similar assemblies to get information about their assembly plan, it is reasonable to use as query a detailed assembly model, so as to take into account all the related information, while, to retrieve models with similar shapes, then it is sufficient to use as query just a list of the parts involved in the assembly. For this reason, this criterion is introduced to specify the **Type of query model**, i.e. how the query is expressed as well as the type of associated input data, and the model **Completeness**, i.e. if all the elements of the query model are defined at the same level.

4.4.1. Type of query model

This criterion indicates the type of data used to represent the query model. According to the reviewed literature, it can assume the following values: single CAD assembly model (*Assembly*), set of CAD assembly models (*Assembly set*), CAD part model (*Part*), set of CAD part models (*List of parts*), and abstract assembly descriptor (*Text*, *Mating graph* or *Graph*).

4.4.2. Completeness

Some works allow leaving unspecified some elements of the query model. In this state-of-the-art, a method that requires the use of a *Complete* query is labeled by the symbol ●, while the symbol ◐is used to characterize methods that can make use of an *Incomplete* query model. This criterion applies to any type of query model, i.e. on both CAD assembly models and abstract representations of assembly models. Naturally, the possibility to make use of incomplete query opens more possibilities to the corresponding retrieval methods.

4.5. Type of similarity

This criterion allows the characterization of the reviewed methods according to their ability to assess *Global*, *Partial* and *Local* similarities as introduced in Section 1. Clearly, the use of an assembly descriptor able to characterize a model at the local level does not imply that the retrieval method exploits this characteristic to assess a partial/local similarity. For example, this happens when the representation of an assembly model is based on graphs and when the matching method looks for graphs or subgraphs isomorphisms. Indeed, a graph-based representation is able to capture a local similarity between two assembly models, this is represented by a common subgraph between the two graphs. If the retrieval method uses graph isomorphism matching applied to the entire query and the entire target models, then the similarity is evaluated at the global level and the local similarity will not be captured. To capture local similarity the use of subgraph isomorphism is preferable.

5. Overview of techniques for assembly model retrieval

While the literature of shape-based retrieval of parts is very vast, the interest of the research community in the retrieval of CAD assembly models is quite recent. In this section, the principal methods for assembly model retrieval are discussed and characterized using the criteria introduced in Section 4. In addition, when possible, the time complexity of the adopted matching algorithm is reported. Unfortunately, since only a few works provide it or detail the adopted algorithms, an exhaustive analysis on the time complexity of the studied methods is not feasible.

Methods have been grouped into two main categories distinguishing the methods that consider only the information related to the parts constituting the assembly model (Section 5.1) from the methods that also exploit topological information among the parts (Section 5.2). Each group is further decomposed in subcategories according to their ability to address the different types of similarity introduced in Section 1, i.e. *Global*, *Partial* and *Local* similarities. This section ends up with a synthesis of the survey and Table 1 summarizes the assessment of each method according to the previously introduced criteria. Obviously, this survey reflects our understanding of the reviewed methods, based on a systematic analysis of the available papers published at the time of the survey.

5.1. *Retrieval methods using only shape information*

This section gathers together methods that address assembly model retrieval based solely on the shape of the constituting parts. The shape description of each part can be managed individually in the assembly descriptor or processed to get a global description of the entire CAD assembly model. Methods are analyzed according to their ability to recognize only global similarity (Section 5.1.1), or also partial one (Section 5.1.2).

5.1.1. *Global similarity*

Renu and Mocko [51,69] explore the use of model similarity and text analysis approaches to develop a relationship between solid models and assembly work instructions. This is aimed at supporting the reuse of decisions taken during the assembly process design. To reach this objective, the authors have fixed three objectives: (i) evaluate solid model similarity in terms of their assembly processes; (ii) investigate the natural language processing approaches required to analyze assembly work instructions; (iii) use part geometry information to mine databases of assembly work instructions and retrieve relevant work instructions. In [51], the authors have faced the first objective aiming to *search for models with similar assembly instructions*, thus addressing the *reuse of product information*. Their process to determine solid model similarity consists of the following four steps:

- **Compute histogram-based similarity scores:** In this step, Osada's method [10] is used to generate shape descriptor for each part of the compared assembly models.
- **Generate clusters of similar solid models based on histogram score**: The adopted shape descriptor provides similarity of overall shape of solid models and it is used to generate clusters of similar models.
- **Compute surface area and tessellation area distribution differences:** Here, to recognize local differences between CAD models, like the one illustrated in Fig. 7, the tessellations of solid models within each cluster are analyzed for evaluating the differences of the surface areas and of the area distributions.
- **Multi-index sorting to generate ranked list of similar solid models**: Finally, a multi-index sorting is performed on the values of the difference parameters (e.g. global histogram similarity, difference in surface area, difference in tessellation area) to rank the models recognized similar to the given query assembly model.

In this work, there is no evidence of the use of assembly relationships information (—). The part information uses a shape distribution (SD), which is computed from the tessellations of the parts. Since the method involves area value in the evaluation of the similarity, the method is scale sensitive (✓). Finally, this method characterizes assembly models at the level of parts and the query model has to be a complete (●) assembly model.

Katayama and Sato [52,53] developed a method for the *retrieval of globally similar assembly models*, which evaluates the similarity according to their hierarchical decompositions. The

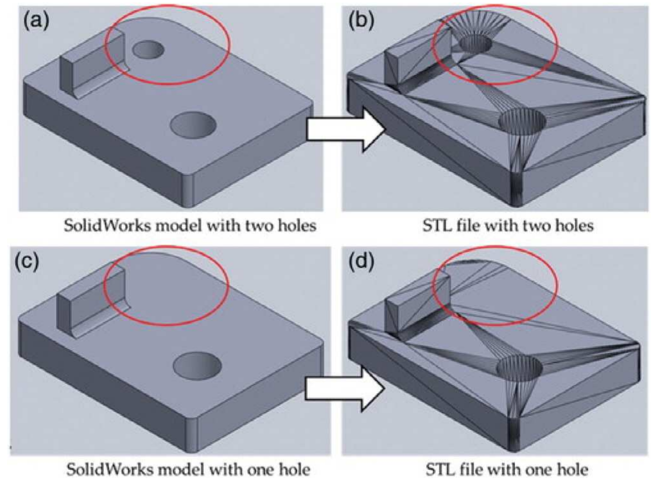


Fig. 7. Local differences in CAD models and their tessellations [51].

idea is to define a new representation of an assembly model, which conveys the global shape of the assembly as well as the shape of the single components. Fig. 8 illustrates the main steps of their method for the similarity evaluation of two assembly models, where different components are specified using different colors. Similarly to view-based methods [8], the components of an assembly model are projected into a set of 2D planes, where the components are identified by their design name. The size of the 2D planes is proportional to the size of the CAD model, and the resulting projections are rotation- and translation-dependent. Then, the 2D Radon transform and the Fourier transform are applied to the results of the projections. The distance between each pair of components is then computed using the Euclidean distance. The final distance of two assembly models is given by the sum of the distances between the corresponding components.

This method characterizes assembly models according to their shape and indirectly to their structure. Indeed, the structure is not considered as a proper topological relation, but it is used to define the way components are projected. For this reason, this method has been included among the works that do not make use of topological information. Anyhow, the fact that the structure is considered means that the adopted assembly descriptor characterizes the models at the assembly level. Parts are characterized by a set of 2D projections (2DP) and the method is scale sensitive (✓). Finally, the input for this retrieval method is an assembly model and the number of components has a strong impact on the similarity assessment, since two assembly models having different numbers of components are not considered similar (●).

5.1.2. *Partial similarity*

To retrieve assembly models, Wang et al. [54] compare the shapes of all the constituting parts. The query is represented by an assembly model but, since only its parts are considered, the query model can be *incomplete* (◐).

In their approach, an assembly model is described as a point set, and the comparison of assembly models is transformed into the comparison of point sets (Fig. 9). The point set is obtained by taking the heights of the bins in the histograms encoding the shape distribution descriptors of the parts of the assembly. In this way, the assembly descriptor is characterized at the part level. Since the same set of parts may originate different assemblies, whose differences cannot be captured, the descriptor characterizes the models at the local level. For the matching of point sets, the *Earth Mover's Distance* (EMD) strategy proposed by Rubner et al. [70] for image retrieval is used. Here,

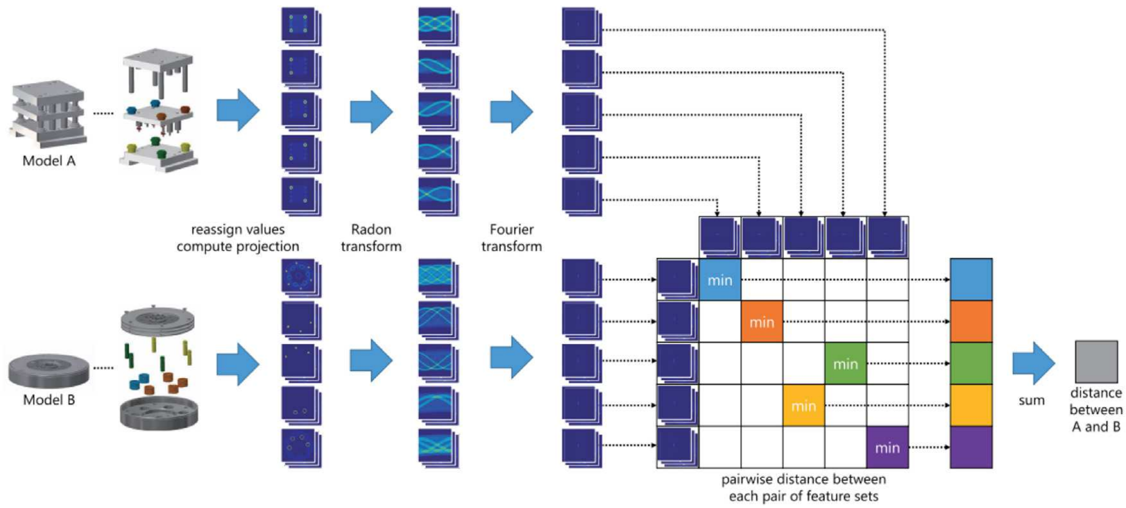


Fig. 8. Example of the procedure to compute the distance in terms of shape and structure between two assembly models [53].

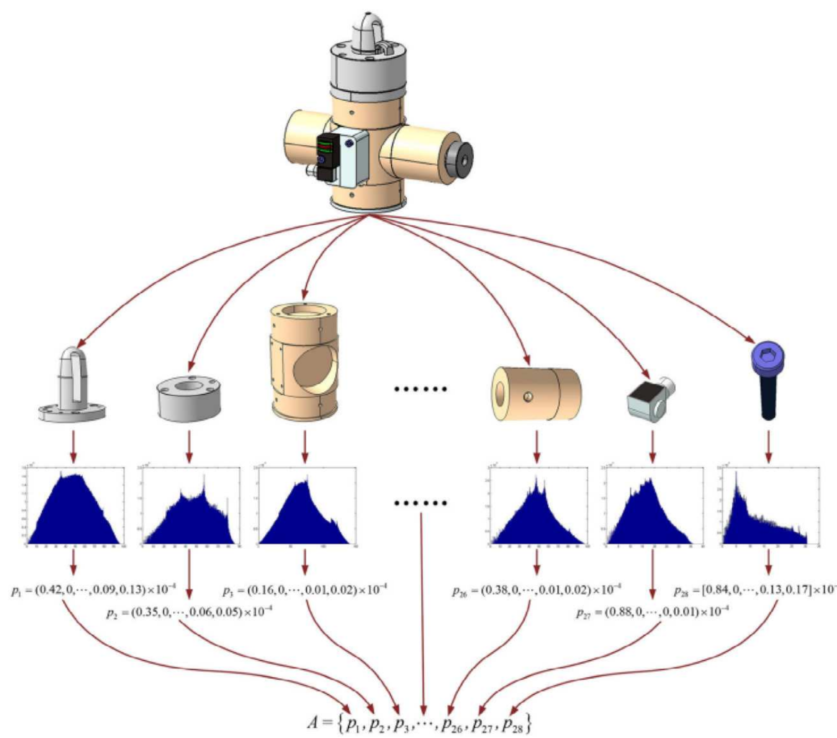


Fig. 9. Example of the description of an assembly model [54].

the complexity of the method depends on the algorithm used to compute the EMD between two n-bin histograms, which *usually* is $O(n^3 \log n)$. Anyhow, they do not provide any further details on the adopted algorithm.

Lately, Zhang et al. [55] adopted a modified Hausdorff distance on the same assembly descriptor. This modified version, proposed by Dubuisson and Jain [71], takes the mean distance between two point sets. In [71], the time complexity is not discussed.

Also in this case, incomplete query models are allowed (◐). Indeed, since the relationships between parts are not considered, the query model can simply be a set of parts which have to be present in the target model. Then, the matching techniques allow retrieving both *Global* and *Partial* similarities. Finally, these techniques suffer from several limitations: (i) the description power of the shape distribution, i.e. complex shapes cannot be disambiguated, (ii) the computational time to perform a many-to-many matching in case of complex assemblies and, above all, (iii) the non-consideration of the relationships between assembly components.

Hu et al. [56] provide two methods for the matching of assemblies exploiting only geometric information of the parts. The first method is based on vector space model (VSM) for the exact matching. In this approach, the generic j-th assembly of their database is represented through a n-dimensional vector $d_j = (w_{1j}, \ldots, w_{ij}, \ldots, w_{nj})$, where n is the number of parts in the j-th assembly and w_{ij} is the weight of the i-th part in the j-th assembly. The weights are computed taking into account two factors. The first relevant factor considers the number of occurrences of a part and the assembly complexity, i.e. the number of composing parts. The second one regards the uniqueness of a part. Indeed, a part that occurs in few assemblies is more discriminating for matching operations. Assembly statistics (PS) are computed by analyzing the assembly and identical parts are recognized by using Light Field Descriptor (LFD) [66]. Using this vector space model (VSM), the assembly descriptor is able to capture local characteristics of the model but not its global shape (or other global features).

The similarity between two assemblies is computed as a function of the angle between their associate VSMs. This approach provides only exact matching, which is too limited in real applications. To overcome this limit, the authors propose also a relaxed matching algorithm. Consequently, the query can be seen as an incomplete assembly model (◐), since it is possible to select an assembly model or just a set of parts that has to be included in an existing model. This matching problem is solved using the graph theory, in particular employing a bipartite graph. The parts of the query and of the target assemblies originate the graph nodes, while the graph arcs represent the similarity between two parts. The bipartite graph matching problem is solved using the Kuhn–Munkres algorithm [72,73] allowing to detect both the *Global* and *Partial* similarities. However, this technique is computationally expensive, $O(n^3)$, thus the authors propose an *approximate matching algorithm*. With their greedy approach, the matching process complexity is reduced to $O(n)$.

The main limitation of the method of Hu et al. relates to the assumption that two assembly models are similar if they mostly share the same parts. This can be a filter to reduce the number of models to be compared, but it cannot distinguish assemblies constituted by the same parts differently assembled.

As a conclusion, in this section, the approaches are strongly based on the shape information and do not use assembly relationships, i.e. no topological information is used. Thus, geometric constraints, kinematic links, or part arrangements are not exploited, which may restrict the application scenarios for which the reviewed assembly retrieval system can be applied. Also, the parts' position is not considered by almost all these works, indeed [51,54,55] and [56] combine only data on the shape of the assembly parts without examining their absolute position. Conversely, Katayama and Sato [52,53] take into account the parts' position in an indirect way. Indeed, they do not reason on the transformation matrices of the parts, but they compute the shape descriptor of assemblies directly on the models resulting from the final assembly process, i.e. considering the parts in their absolute position.

Lastly, the methods using assembly descriptors able to characterize assembly models at local level fail to identify local similarity because of the adopted matching techniques. Effectively, there is no evidence that the query model may be bigger than the retrieved correspondences, then they require that the query model is included in the target model at least. Theoretically, by revising the matching procedure, these methods can detect also local similarities.

5.2. Retrieval methods using shape and topology information

This section presents retrieval methods that make use of assembly relationship information. The relationships in assembly models may be represented by the use of graph-based descriptors. Even if different graph structures can be adopted, individual assembly components are usually represented as nodes, links between components as arcs of the graph and other information are represented in form of attributes of nodes and arcs. Here again, methods are categorized and analyzed according to their ability to recognize global, partial and local similarity and finally those that assess all the three different types of similarity.

5.2.1. Global similarity

Tao and Huang [57] propose an approach to find assembly models for design reuse and generation of manufacturing plans. Their *component attributed relational graph* (CARG) represents an assembly model as a direct graph where the nodes represent the components and the arcs correspond to connections between two components. Several attributes are associated with the nodes and the arcs for the description of the assembly model. In particular, each node encodes the component volume, the surface type (SI), the surface convexity (SI), the loop number of a face (LN) and the edge number of its outer loop (EN). An arc represents the adjacency relationship between two components and encodes the types of contact surface pair and the connection relations, which can assume the following values: screw connection, pin joint, key joint, rivet joint, bearing, belt, chain and bonding or welding. Since the description of the single parts has been preserved in each single node, and has not been collapsed into a unique assembly descriptor, this descriptor is able to capture both the entire assembly as well as the details at the level of its parts.

Using a graph as assembly descriptor, in principle, local similarity could be detected. However, the adopted matching procedure is based on a global evaluation of the similarity. It computes the similarity $S(P^1, P^2)$ between two components P^1 and P^2 considering the stored properties and the connection relations. To evaluate the similarity between two assemblies A^1 and A^2, a compatibility matrix $SM(A^1, A^2)$ is built, where the element in the i-th row and j-th column is $S(P_i^1, P_j^2)$. Then the similarity between the assemblies A^1 and A^2 is computed as follows:

$$S(A^1, A^2) = \frac{SM(A^1, A^2)_{max}}{\max(m, n)}, \qquad (1)$$

where $SM(A^1, A^2)_{max}$ is the value of the optimal matching compatibility matrix $SM(A^1, A^2)$ evaluated using Kuhn–Munkres algorithm [72,73] and m (resp. n) is the number of components in A^1 (resp. A^2). The authors do not discuss the complexity of their method, anyhow the Kuhn–Munkres algorithm usually has $o(k^3)$ time complexity [74], where, in this case, $k = \max(n, m)$.

This method uses simple geometric information and kinematic links to characterize parts of assembly models and their relationships. The geometric information is partially directly available from the B-Rep model of each part (A), and *Kinematic links* are available likewise (A). The use of surface area makes the method sensible to dimension differences, i.e. it is scale sensitivity (✓). The values as "rivet joint", "bearing" and so on, used to characterize the connection relations, suggest that some attributes (✓) are supposed to be available in CAD models.

Miura and Kanai [58] provide a 3D shape retrieval method which satisfies the following four requirements: (i) evaluating assembly structure similarity (i.e. the method evaluates shape and structure similarities of the assembly); (ii) maximum matching ability (i.e. similarity measure should not change significantly if a minor component changes in the assembly); (iii) insensitivity to the movable components (i.e. similarity measure should not consider relative positions of the components in an assembly model); (iv) flexible control of similarity evaluation (i.e. similarity should be easily defined and tuned by the designer).

The adopted assembly descriptor is an assembly graph, where each node corresponds to a component in the assembly and each arc indicates a contact, an interference or if at least a geometric constraint is present between two components. To characterize the geometry of the model, several attributes are specified for nodes and arcs. In particular, the area, the volume and the angle distance (AD) are associated with nodes, while the type of constraints characterizes the arcs (if they identify a constraint). The data that characterize the constraint information has to be available (A) in the CAD models. The use of a graph representation guarantees a local characterization of the assembly model, anyhow the assembly retrieval is performed by a global graph matching, that it is treated as the *stable marriage problem* and solved by the Gale–Shapley algorithm [75], which usually runs in time $o(n^2)$ [76], where, in this case, n refers to number of nodes of the graph representing the query model. Even if, there exist several optimizations to reduce the complexity for this algorithm, the authors provide no information about their use. The similarity is evaluated at the level of the components by using the difference between the stored shape features of two components and at the level of the structure by using an assembly graph matching. The method is scale sensitive (✓) because of area and volume information used in the matching procedure. Finally, the considered "assembly structure" refers to the kinematic links and constraints between two assembly parts and not to the hierarchical organization of an assembly model.

The two methods presented in this section identify assembly similarities at a global level, even though the adopted descriptors are able to characterize assembly models at a local level. It means that adopting other matching approaches, these methods could theoretically evaluate assembly similarities also at partial or local levels.

5.2.2. Partial similarity

With the aim of improving the product conceptual design reuse, Han et al. [59] provide a method to reuse knowledge design through the retrieval of assembly models according to high-level of semantic knowledge. Differently from all the methods examined in this review, this approach does not use a shape description to represent the constituent parts but it strongly promotes semantic information derived by text annotations or component names.

In this approach, an assembly model is described by three different semantic information: *part semantic*, *assembly constraints semantic* and *functional semantic*. Part semantic includes information such as assembly name, part name, part type (e.g. function or connector) or material. Then, this method makes a large use of annotations (✓). Assembly constraints semantic describes *geometric mating* (i.e. the geometric constraints as coincident, concentric, distance, tangent, parallel and so on) and *connection relationships* (i.e. hard connections that are realized by physical connectors as screws, nut or pin, or soft connections). This method makes use of topological information: geometric constraints with the use of geometric mating, and kinematic links since connection relationships are interpreted to deduce the degree of freedom between two parts. The authors state that parts and constraints information is parsed into standard semantic terms, then these data are partially computed (PC) because they are supposed to be available in the CAD model or perhaps extracted by using other existing approaches. Finally, functional semantic classifies eight basic functions, i.e. support, signal, channel, connect, control, convert, provision and branch. The functional semantic annotation is an automatic process that exploits the two previously introduced semantics. Then the functional classification is achieved by algorithmic reasoning (AR). The information on the parts combined with the information on the constraints allow describing the assembly model at both global and local levels.

A novelty of this method relates to the query input, i.e. a text query that describes the component, the constraints and the functional semantics to be matched in other assembly models. From the proposed examples, it emerges the possibility to provide incomplete queries (◐), i.e. the query semantics include part semantics, assembly semantics and functional semantics, but the user can specify a combination of these elements leaving unspecified some of them. Once a semantic representation is created for the query and the target models, a semantic similarity evaluation is performed by using WordNet [77] and ontology. WordNet is an English dictionary used to handle synonyms, while ontology can be viewed as a directed graph [78]. Then the similarity evaluation is translated into a bipartite graph matching problem that is solved using the Kuhn–Munkres algorithm [72,73], which usually runs in time $o(n^3)$ [74], where, in this case, n refers to the maximum number of nodes of the graphs representing the models to compare. Finally, this matching technique allows to detect both the global and partial types of similarity.

Li et al. [27,60] address the reuse of previous solutions in the design of new products to avoid starting from scratch. They aim to define a geometric reasoning approach independent from any CAD system or design history. Their method has been conceived for CAD parts, but they include a generalization for assembly models as well. In [27], they exploit a hierarchical representation of CAD models, which is composed of a tree-like structure (TR) that describes the global similarity, and an adjacent graph (ADJ), which characterizes local similarity. In this way, the method can support the assessment of global and partial similarities. Using this scheme, the root of the TR represents the entire model, the intermediate nodes represent a set of partial features and the leaves are associated with detail features, e.g. a single face of the solid model or a more detailed partition of surfaces. While, the ADJ encodes relationships between non-leaf nodes, for the parts it defines if two features have a common edge. This representation can be used also for assembly models, anyway the authors do not suggest directly to use this method, because kinematic information is not extracted and described explicitly. The approach has been applied to assemblies to support the reuse of mold designs [60]. In case of assemblies, the TR corresponds to the assembly hierarchical decomposition, while the ADJ captures kinematic pairs between parts. The similarity is computed by a subgraph isomorphism on the ADJs, which is based on the VF2 algorithm [79], which usually has time complexity of $o(n^2)$ for the best case and $o(n!n)$ for the worst one [80], where n refers to the maximum number of vertices of the two graphs. Then, the similarity is evaluated for each level of the TR in terms of their shapes and relationships. In particular, the shape similarity S_i in the ith level is evaluated by the following equation:

$$S_i = \sum_j \omega_j \times D2_{sim} \tag{2}$$

where ω_j is the ratio between the area of the jth matched pair in the ith level over the area of all the matched pairs in the ith level and $D2_{sim}$ is the similarity between D2 shape distribution (SD) of matched pairs. To improve the efficiency of the retrieval, a coarse filtering is performed in advance. Parts are classified in six types of models (beam type, block type, round type, cover type, case type and frame type) based on KNN algorithm [81]. Thus, this method exploits a functional classification performed by an artificial intelligent technique (AI).

The way kinematic links are processed is not detailed, thus one can assume they should be available (A) in the CAD models. The use of the area of matching components in the similarity evaluation makes this method scale sensitive (✓). In the end, it allows global and partial similarities assessment, but both are achieved by exploiting the hierarchical structures (✓). This suggests that the method is not able to recognize as similar two models using different sub-assembly organizations to represent the same product.

Deshmukh et al. [47] have proposed a retrieval method exploiting outcomes of previous researches [61,62]. In these works the authors describe a system for extracting information useful for searching and retrieving assemblies from databases. The functional data are not usually explicitly stored in CAD files and often they cannot be inferred from the geometric characteristics of the assembly either. Thus, in this system, the functional classification, if not present in the CAD files, has to be specified by the user (US). The authors take into account many aspects that play a meaningful role in the description of an assembly model. The data structure used in this work is a mating graph. For each assembly in the database, its mating graph is built, where each node corresponds to a part in the assembly model and each arc represents a mating condition between two parts. In particular, for each part in the assembly model, they consider the following information.

- **Category:** It describes if the part is *standard* or *custom*.
- **Geometry:** The authors specify two types of geometric information depending on the category of the parts. If the part is standard, then its geometric information is referred by a library of standard parts, otherwise, it is referred by the best approximation in the assembly database returned by the approach described in [82], which is based on Face-based Attributed Applied Vectors (FAAVs).
- **Type:** This information is defined only for standard parts and indicates the subcategory of the standard part.
- **Degree:** Given a part *P*, its degree represents the total number of parts that are in contact with *P*.

Category and type can be unspecified if not provided by the designer. Considering part relations, the user has the possibility to select the types of relation between two parts or to leave them unspecified. This allows defining also incomplete queries, where not all parameters are defined (◐). The query model is represented as a mating graph, where parts are represented as nodes with the four above mentioned attributes (category, geometry, type and degree) and if two parts are in contact then an arc exists between the two corresponding nodes. Most of the information used by the authors to characterize parts are related to attributes (e.g. category and type), while the geometric information characterize parts by using their B-Rep representation to compute area, curvature distributions (C) and the type of surface of the part faces (SI). These data are available in the B-Rep of each part, but the assembly relationships are not always present.

The proposed algorithm for mating graph-based search is addressing on *graph compatibility problem*. It is performed through a combination of several heuristic approaches. To improve the results, the search space is reduced assigning a priority score to each node in the query graph. Then the algorithm attempts to match each node from the query graph to a node from the database graph by recursive operations and visiting the graphs by a depth-first search. This procedure allows addressing partial similarity but not the local one. In addition the partially similar assemblies are retrieved using the assembly structure.

Chen et al. [48] propose an approach exploiting the hierarchy in product structure and the semantics of assembly interfaces. In addition, they also provide indexing and filtering mechanisms. The assembly descriptor presented in this work is a graph, which takes into account different information levels. It includes topological and geometric information of the assembly. More precisely, the descriptor comprises the following information:

- **Topological structure** illustrates how the assembly, the subassemblies and the parts are connected together, and it also includes the hierarchical assembly structure.
- **Assembly semantics** describes the type of the relationships between the parts in an assembly model through the so-called semantic assembly interface. It is defined as a multilevel information involving function layer, implementation layer and geometry layer. The function layer considers the degrees of freedom (DOFs) between two connected components in the assembly. It counts the number of translational, rotational or composite (i.e. the combination of multiple DOFs together as the screw joint) degrees of freedom between two components. The implementation layer defines and counts the types of kinematic relations between two components. In the end, the geometric layer contains information about the geometric-matings typically used in assembly modeling that can exist between two components sharing contacts, e.g. if they are concentric, perpendicular, parallel or at a fixed distance.
- **Geometric information** is used to describe the shape of each assembly component. It is stored in the corresponding nodes. In particular, if the component is a part, its shape distribution vector is computed. If the component is a subassembly, then the shape distribution vector of the bounding box of the component is computed.
- **Attributes** allow to consider other data, such as the functions, i.e. the task that a system or a component is able to perform; the loads, i.e. the forces, deformations or accelerations applied to a structure or its components; and the environmental conditions.

Thus, this approach makes use of shape distribution (SD) as shape descriptors for the parts, and it exploits the structure, the kinematic links and the constraints to characterize the assembly. The geometric information is computed, while the structure is read (A) from the CAD model. To the best of our knowledge, this work is one of the first assembly retrieval approaches, which tries to identify kinematic pairs automatically. Anyhow, the authors state that some complex kinematic pairs need to be inserted manually (PC). The authors recognize the importance of semantic information (e.g. functional component classification), which is therefore supposed to be manually specified by the user (US).

Since the assembly descriptor contains numerous data, the matching procedure is divided into two main steps to simplify the whole retrieval process. The first step takes into account the topology structure of the multi-level assembly descriptor. The hierarchical graph matching is carried out using the VF2 subgraph isomorphism algorithm [79], which has time complexity of $o(n^2)$ for the best case and $o(n!n)$ for the worst one [80], where n refers to the maximum number of vertices of the two graphs. Additional information is used to prune the matching algorithm. For the algorithm, two nodes are equivalent if the query node has fewer children than the compared node; while two arcs are equivalent if they have the same DOF. The second step refines the obtained matching by considering the assembly semantics and the geometric information. This last step evaluates also the arrangement of the assembly components in the 3D space using an "assembly bone" representation, i.e. a structure composed of line segments which connect the geometric centers of two components. Here, the part arrangement information is not explicitly stored in the assembly descriptor but deduced (C) and used during the matching process. The use of this type of matching supports partial retrieval if the two compared assembly models have the same structure, i.e. only if the query model is present in the target model as subassembly. This limitation could be overcome, by slightly changing the process at the price of additional computational time, still leaving the user the possibility to select how important is the structure for his/her purpose.

5.2.3. *Local similarity*

Zhang et al. [63] have been working on the retrieval of CAD assembly models to reuse the embedded design knowledge and to improve design efficiency when developing new products. They propose a generic face adjacency graph (GFAG) to discover common design structures automatically among assembly models.

The GFAG can capture the geometric and topological information of an assembly model. Therefore, it is suitable for assembly characterization, where the relationships between components are encoded through the concept of mating face pair (MFP). In this graph, nodes correspond to parts of the assembly model and edges correspond to the MFP between two parts. Each part is represented by a face adjacency graph (FAG) [83], where nodes correspond to faces of the B-Rep of the part and arcs correspond to adjacency edges between faces. Faces are then classified as planar, convex, concave or transition according to the sign of the principal curvatures computed for sample points. Similarly, edges are classified as concave, convex or smooth depending on the dihedral angle formed at the edge's sample points by the tangent planes of the faces sharing the edge. Thus, considering the criteria of Section 4.2, this descriptor uses geometric information, in particular it is based on surface curvature (C), dihedral angle (DA) and surface information (SI). The relationships and kinematic links between two parts are assumed to be present in the model created with a commercial CAD systems (A) and can have the following values: coincident, contact, offset and angle. A shape vector descriptor is computed for each part in an assembly model using sampled points of the FAG of a part and the ones of its mating parts. In this way, the description of a part is influenced by its contacts, thus it changes if different parts surround it.

Since, this last characteristic can reduce the portion of common structure detected (i.e. local similarity), the authors have extended their work [64] to provide a graph descriptor that describes independently parts and mating relationships of an assembly model. Parts are represented by vectors of shape distributions and contacts are quantified by the following equation:

$$L_p = \frac{S_p + S_{p'}}{2} \tag{3}$$

where the vector L_p represents in a single relation the multiple contacts between two parts p and p', whose shape vectors are S_p and $S_{p'}$. Thus, in this revised approach, the descriptor uses shape distribution (SD) and the available kinematic links (A). No attribute is encoded and, since no normalization of shape distributions is mentioned, the used information varies under scaling operations and the descriptor is scale sensitive (✓). Besides the fact that the mating information has to be available (A) in the assembly model, there is no information about the semantics of the relation between two parts, i.e. the type of the involved contacts. This is a limitation since it does not fully characterize assembly models regarding how parts are connected. Indeed, depending on the nature of their contacts, the motions between two parts can differ.

Lupinetti et al. [65] propose a method which fully characterizes an assembly model to assess all the three types of similarities between assembly models described in Section 1, i.e. global, partial and local similarity. Their approach is based on a multilayered description, called Enriched Assembly Model (EAM), which encodes four different types of information concerning the structure, the shapes, the interfaces and the statistics of the assembly. The structural information layer encodes the hierarchical sub-assembly organization as specified by the designer, the *type of component* (e.g. screw, nut, gear, shaft) and information on *parts' arrangement* (i.e. the regular patterns of repeated parts). The considered patterns collect the maximum number of repeated parts whose barycenters are equidistant and lying on the same linear and circular paths. To overcome the possible absence in CAD models of the information on component type and on patterns, the authors provided tools for their extraction [84–86]. The shape layer describes parts in terms of their 3D spherical harmonics [68], their volume and their surface area, which scale are sensitive (✓). The interface layer encodes the kinematic links, which describe the type of contacts and degrees of freedom between two parts. Finally, the statistics layer contains numerical attributes to allow a quick search and filtering, e.g. the number of patterns of a specific type.

These data are organized into an attributed multigraph-based representation allowing to describe and to characterize components at the levels of the parts and of the assembly at local as well as global levels. All the data necessary in their proposed assembly descriptor are automatically extracted exploiting only the information available in the CAD models and engineering knowledge; no meta-data neither attributes are used (−). In particular, assembly topological information is extracted by reasoning on the geometry of the parts (C), while functional classification is achieved by using a supervised learning algorithm able to provide a shape-based classification further refined with the use of engineering rules to interpret common design practices (AI). Then, all the data are computed except the sub-assembly structure that is supposed to be available (A) from the CAD models. Fig. 10 illustrates an example of the multigraph created from a CAD model. For readability purposes, only a few attributes are shown: single line-circled nodes indicate single parts, double-line nodes (S and N) designate a set of parts belonging to circular rotational patterns, node labels denote the type of component, straight arcs connect two parts in contact and the associated label indicates the DOF, finally wavy arcs represent curve contacts.

In addition, this approach provides the possibility to specify an abstract query (◐) defining the constituting components and the related interface links by a graph-based description supported by a dedicated user interface. Moreover, the user can select the set of criteria according to which the assembly should be similar. Thus, the complete EAM is computed for the models in the queried dataset, while for the query model, only the layers containing the specified criteria are computed and exploited for the matching (◐). The similarity between two models is detected solving a Maximum Common Subgraph (MCS) matching problem [87] on an association graph created by putting in relation the nodes and the arcs of the compared EAM according to the selected criteria. The MCS is managed as a Maximum Clique (MC) detection problem with the use of the simulated annealing method [88]. For simulated annealing, the time complexity is usually $o(n^2 \log(n))$, where n corresponds to the number of nodes in the graph, which, in this case, is an association graph that in the worst case is $n = n_q n_k$ where n_q and n_k correspond to the number of nodes of the graphs representing the models to compare. With this technique, it is possible to evaluate all the three types of similarity, i.e. global, partial and local, without using the hierarchical structure of the assembly. This means that the proposed method is able to identify as similar two assembly models which represent the same object but with different structures.

In [89], the authors, differently from the other existing methods, also investigate how to visualize the retrieved models in a convenient manner. As a result, the system highlights the different types and criteria of similarity, to allow a better comprehension of the results and thus a fast identification of the target models. This is an interesting research topic arisen by the needs of comparing models under multiple perspectives.

5.3. Synthesis of the state-of-the-art review

Table 1 summarizes the approaches that address CAD assembly models similarity evaluation and which have been discussed and characterized in this survey. From the analysis of these works, it can be observed that almost all of them assume the full availability of the information necessary to derive their assembly model descriptors. This may be an important limitation since not all the necessary data are present in the CAD models. Indeed, supposing users can add all the missing data is not reasonable, as it

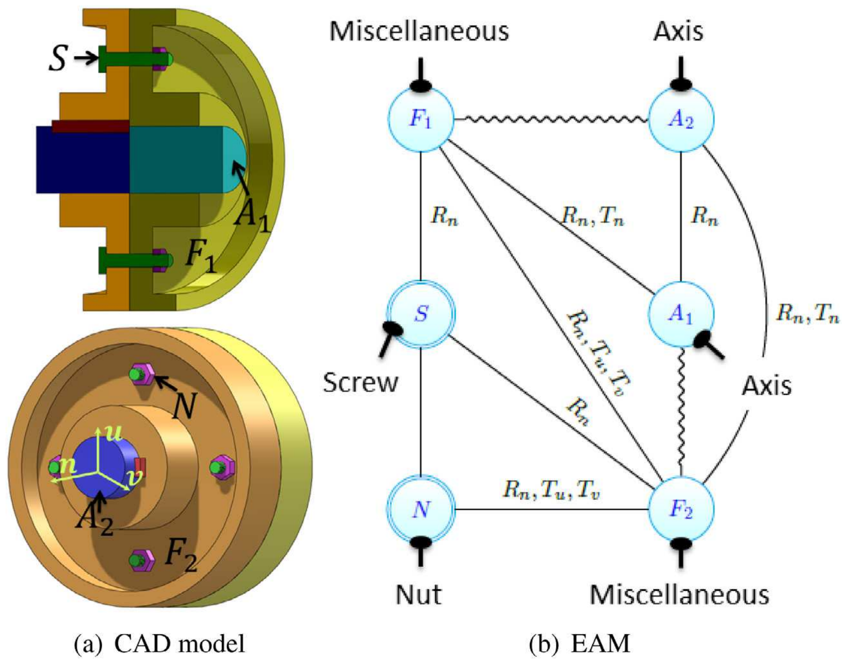


Fig. 10. Example of a CAD model and a part of its EAM descriptor [65].

is time-consuming and tedious for the user. Unlike what happens for the characterization and recovery of individual parts, no work uses the shape features to characterize assembly models. For this reason the Feature criterion does not appear in Table 1. Arguably having an assembly retrieval method able to manage this type of data would greatly increase the complexity of the system.

In general, the research regarding the characterization and the retrieval of CAD assembly models is still in progress and many issues are still to be addressed. Most works address the problem of retrieving models partially similar, which anyhow have solutions only taking into account the hierarchical structure. However, the limitation is that the similarity of the models' structure becomes a constraint and the user cannot retrieve an assembly model included in a bigger assembly if the query is not represented as a subassembly in the target model. This hypothesis can affect scenarios whose purpose is the maintenance of assembly components. Indeed, in this situation, a component included in an assembly model should be identified despite its designed structure.

Few works address different types of similarities between assembly models but some of them could probably be extended in the future to evaluate local similarities. Usually, the geometry, the size of the components and the different types of their relationships are used to compare CAD assembly models. However, in several approaches the extraction of these data is not faced and the information is supposed to be available or added by the user. Moreover, the practice of characterizing parts by their shape does not allow to treat the possible simplified descriptions of components in assembly models as discussed in Section 2.1.

6. Future challenges

Today, the existing approaches already help designers to improve the efficiency of their work within the PDP by facilitating the access and the exploitation of existing knowledge. Despite those achievements, further efforts are still necessary to make available tools allowing a more powerful retrieval of CAD assembly models according to different similarity criteria. In the following, some of the challenges that still have to be faced to reach this goal are discussed.

6.1. *Conjugation of semantic and geometric information*

Using current tools, it is possible to perform searches using PDM/PLM systems [90,91] and then to analyze the content of the filtered models by using retrieval techniques. This two-steps process must be improved in the future through a tighter integration by developing a retrieval system based on a descriptor that uses both geometric data of the CAD models and semantic information present in the PDM/PLM systems. The simple merging of the two types of information is not a big deal, the real challenge is the ability to extract the implicit knowledge that may derive from the combination of the two types of information. For instance, a volumetric intersection between two components in a digital model cannot exist in the corresponding real components, then it should be solved before components production. Anyhow, sometimes volumetric intersections are designed on purpose, for instance when the intersection involves a deformable element, such as seal ring, gasket or rubber washer. Currently, engineers perform this decryption by analyzing the information separately. To support the automation of the processes demanded in the framework of Industry 4.0, it is crucial to have tools able to complete such retrieval tasks in a completely automatic way and to assist designers in the decisional process. Designing new processes able to exploit the combination of semantic and geometric information is challenging and requires many contributions to be achieved, such as the interpretation of the assembly relationships (Section 6.1.1) and the functional classification of the components (Section 6.1.2).

6.1.1. *Interpretation of assembly relationships*

Relationships among components are the backbones to characterize CAD assembly models. Actually, almost all the works addressing assembly retrieval are based on the hypothesis that these data are available even if in most cases they are not. Luckily, several works exist for the extraction of assembly contacts in different application domains. Yang et al. [92] proposed a simulation system for assembly process based on constraints recognition. Park and Oh [93] developed an automatic method to extract kinematic information from assembly models and Lupinetti et al. [94] proposed a method for the detection of the mating relationships, the computation of their degree of freedom and the equivalent joint resulting from all the contacts between two parts.

Besides the extraction of assembly relationships, other methods have been proposed to identify and exploit the meaning of these relations. Swain et al. [95] have defined an extended liaison (joints) to integrate the information between the product model and the assembly process. This complex approach allows the identification of the assembly process of riveting, welding, screw fastening, bolt fastening and gluing. For finite element analysis, Shahwan et al. [96] described a qualitative reasoning process to detect and to functionally classify component interfaces. Their method is based on the definition of conventional and functional interfaces. It starts from the CAD assembly model and exploits additional knowledge expressed in an ad-hoc ontology.

Despite these first attempts, the research on the interpretation of assembly relationships must still face important issues. Among the possible relationships, clearances between two parts can convey an engineering meaning or can be originated by errors. Since they do not identify always unrealistic arrangements as for the volumetric interferences, distinguishing the correct and incorrect configurations requires specific reasoning. Some reasoning can be done combining the distance between the faces of two parts and also the types of involved components (e.g. screws, nuts, brackets, keys and spacers) to find out if the clearance is a "real clearance" or if it represents a missed contact. In addition to the component type, this problem could be faced considering also information about the part features. Indeed, the characterization of holes, pockets, slots and fillets may help in the clearance interpretation.

6.1.2. Functional classification of assembly components

Recognizing the functionality of a component can be challenging because components with the same functionality can have different shapes and vice-versa, then the use of multi-source information of different nature can facilitate the automatic classification of 3D models.

In the mechanical field, Ip et al. [81] define a feature space where they apply decision tree learning and reinforcement learning to classify solid models. This method allows the automatic classification of wheels, sockets and housing models. In [97], the authors present an automatic model classifier for CAD models integrating machine learning techniques. Using a series of shape descriptors, their approach aims at learning multiple CAD classifications and is applied to the classification of prismatic machined parts and parts with finishing features machined after part casting. The classification proposed by Pernot et al. [98] also exploits a series of shape descriptors and classifies products in terms of characteristics that can affect the simplification process for the Finite Element Analysis (FEA) of parts. Hence, their categories are: thin parts, parts with thin portions and normal parts. Qin et al. [99] present an automatic 3D CAD model classification approach based on the deep learning technique. Their method considers 28 different functional classes and combines different training strategies to simulate engineering manual classification processes.

However, as highlighted in [100] and [101], the functional classification of 3D models requires information on the context of use of the parts. Shahwan et al. [43,96] analyze functional interferences from the geometric interferences of parts in an assembly and identify functional designations, as cap-screw, tubular rivet, gear. The main limitation of this method is the complete entrustment in the design methodologies. The extension of this work [44] uses mechanical equilibrium state analysis for assigning only one functional interface to several geometric interfaces. The approach is semi-automatic and users have to identify the start and the end of the kinematic chain in the assembly model. Recently, Lupinetti et al. [86] present a preliminary work based on a multi-step approach, which first assigns a category to each part according to some shape characteristics and exploiting a machine learning technique, then it assesses the initial classification by analyzing the context of use of the part in the assembly. In this way, the authors try to overcome problems due to idealized designs that make challenging the identification of the components.

6.2. Definition of assembly similarity

Assembly models may be considered similar under various and different criteria (e.g. global shape, kinematic links, component dimensions), moreover different types of similarity can be fulfilled (i.e. global, partial and local). To the best of our knowledge, the majority of works, which allow to retrieve assembly models according to different criteria, combines shape similarity criteria based on the components with relationships criteria based on the assembly. However, few works allow combining the functional aspect of similarity (e.g. power transmitter) of the assembly components together with geometric characteristics (e.g. round shape). Finally, few works address local similarity and those that face partial similarity strongly rely on the hierarchical assembly structure, i.e. two models can be considered partially similar only if one is present as a subassembly in the other.

Actually, considering the results achieved until now and the numerous scenarios where retrieval frameworks are useful, an interesting challenge still remains and is represented by the research of *what is similar?* and *why?*.

6.3. Specification of the query

A useful feature of retrieval systems is represented by the chance of querying databases using different descriptions for the query model. This practice has already been studied for the parts model, where there exist systems that allow retrieving 3D digital models by providing a 2D freehand sketch [102–105]. Considering assembly models, this functionality may be useful, for instance, both in the early design stage and considering the scenario of reverse engineering, when probably a designer may not have an available CAD model of what he/she wants to retrieve. In these contexts, a 2D drafting or a scanned model respectively can be more appropriate as query input of the retrieval system. Being able to retrieve CAD models similar to a given point cloud could be of great interest in the scope of the Industry 4.0, and notably when developing Digital Twins.

Moreover, it is important to allow specifying similarity criteria in an intuitive way. To this purpose, some works address this topic giving the possibility of using query model partially defined or abstract query model not originated from an existing CAD model, while another one offers a semantic definition of the query. Anyway, these new ways to specify queries are still in their infancy and have not been adequately studied. In the future, it is desirable to develop more intuitive interfaces to meet the designers' habits.

6.4. Development of an efficient system

The explosion of digital data has made available 3D CAD models to designers offering the possibility of reusing existing solutions. Anyhow, the size of the digital models (also in terms of the constituent part number) and of the databases make challenging providing an efficient retrieval system that properly satisfies users' needs. To this purpose, retrieval methods often split the comparison procedure into two steps: a primary similarity assessment extracts candidate models, while a second refinement

Table 2
Characteristics of the datasets used to evaluate available assembly retrieval approaches.

Approach	Number of assemblies	Number of parts	Number of unique parts
[52,53]	15	–	–
[54]	409	6315	–
[55]	160	1135	–
[56]	614	5100	2814
[59]	502	6348	–
[61]	200	–	–
[48]	2249	10,062	–
[65]	140	15,057	5343

improves the results of the previous retrieval. This practice is a first effort to provide an efficient retrieval system to the final user, but it is not enough, especially since many approaches of the literature are tested on small datasets, as illustrated in Table 2, whose sizes are not comparable with those of real company databases.

Thus, the development of an efficient system is still an open issue.

6.5. Evaluation of the system effectiveness

Important limitations exist to evaluate and compare the search effectiveness, i.e. the ability to retrieve the maximum number of relevant models within a limited number of retrieved models. Generally, the effectiveness of a system can be evaluated by *precision* (the number of retrieved relevant results over the number of retrieved models) and *recall* (the number of retrieved relevant results over the number of relevant models in the dataset) measurements. An ideal system should have the maximum values for these two measures, but generally, they are inversely proportional, i.e. when the recall increases the precision decreases. Anyhow, not all the considered methods provide this study and most of them are tested on different datasets not allowing to properly compare and assess their effectiveness. So far, differently from what exists with simple parts [101,106,107], an assembly ground-truth (or more simply a CAD assembly repository), whose content can be used as benchmarks, does not exist. This lack derives from the difficulty both to get realistic models and to label target models properly as relevant or not relevant according to a given query (especially in case of partial and local matching). To facilitate the target issue, Chen et al. [48] assumed that a model is relevant according to a query if there exists a subassembly in the target model similar to the query model. This practice can ease the creation of an assembly benchmark, but it raises the issue of the models considered as false-positive, i.e. two models that represent the same object can be considered one relevant and the other not relevant simply according to their hierarchical structure. Anyhow, this practice can be reasonable only if the user specifies the structure as a similarity criterion.

These difficulties make challenging the definition of a general ground-truth, which is essential in order to compare the existing retrieval systems.

6.6. Visualization and interpretation of the results

Last challenge regards how to visualize the results of a given query highlighting the different similarities to the user in an intuitive and clear manner. This aspect becomes crucial when dealing with partial and local similarities, while it is less essential in global retrieval. It becomes even more challenging when the retrieved objects to be analyzed are complex assembly models. Many works considered in this review test their methods on assemblies made up of a relatively small number of parts (approximately 10 parts), then in this situation, the visualization of the results is not challenging. Anyhow, in real industrial configurations, the visualization of the results should highlight the multiple information that characterizes the identified similarities.

In the future, results visualization could exploit VR/AR/MR technologies to provide enrichment of the perceptual information while browsing in a set of 3D models. For instance, if a user searches similar components to be replaced in an existing object, then by the use of mixed reality he/she can virtually add the digital retrieved solutions in the real object and evaluate the replacement directly.

7. Conclusion

Accessing previous knowledge associated with existing products and past realizations may drastically improve the efficiency of the entire product life cycle, from its conception up to its disposal. This is a key issue to develop the competitiveness of companies worldwide. However, knowledge about a product changes dynamically throughout its life cycle stages, then knowledge-based searches are still very challenging as it has been highlighted in Section 6.

This paper analyzes the state-of-the-art of the approaches and systems for CAD assembly models retrieval clustered according to some criteria that allow to better highlight the main features of the proposed works and to discuss the open issues. The criteria reflect the characteristics we believe important to satisfy the needs of the identified usage scenario of the retrieval system. Effective assembly retrieval systems need to consider the multifaceted information characterizing an assembly, which is not always explicitly available and thus has to be extracted through automatic reasoning processes. Various works in literature are addressing these issues to some extent; however, they take into consideration only some configurations and product types. Therefore, additional efforts are still needed to create a fully functional automatic system dealing with all the possible configurations. In particular, limits currently exist in interpreting some types of relations among components, i.e. the clearance and volumetric interactions. In addition, detection of functional sets in assemblies, which can be themselves a search key, is still limited to specific product categories.

Efforts should also be devoted to scalability and efficiency. Current systems have been generally tested on limited sets of assembly models consisting of a relatively small number of components and almost no one is paying enough attention to the indexing aspect, which can play a critical role in case of a large dataset with huge models made of several hundreds or even thousands of components. In addition, the lack of a proper dataset of CAD assemblies prevents a reliable evaluation and comparison of the effectiveness of the existing retrieval systems: not all the works considered include an evaluation of the effectiveness of the proposed method; furthermore, the methods are however tested on different datasets, normally not available, thus preventing a reliable comparison. Finally, the lack of well-organized datasets of CAD assembly does not allow the adoption of deep learning approaches in the matching process, which seem promising when compared to traditional methods.

In conclusion, even if some important results have been achieved, further research and development are still needed to define an efficient and comprehensive retrieval system fully satisfying the range of assembly retrieval usages in the whole product development process.

References

[1] Mahdjoub M, Monticolo D, Gomes S, Sagot J-C. A collaborative design for usability approach supported by virtual reality and a multi-agent system embedded in a plm environment. Comput Aided Des 2010;42(5):402–13.
[2] Erohin O, Kuhlang P, Schallow J, Deuse J. Intelligent utilisation of digital databases for assembly time determination in early phases of product emergence. Procedia CIRP 2012;3:424–9.
[3] Chandrasegaran SK, Ramani K, Sriram RD, Horváth I, Bernard A, Harik RF, et al. The evolution, challenges and future of knowledge representation in product design systems. Comput Aided Des 2013;45(2):204–28. http://dx.doi.org/10.1016/j.cad.2012.08.006.
[4] Lu Y. Industry 4.0: a survey on technologies, applications and open research issues. J Ind Inf Integr 2017;6:1–10.
[5] IBM. Speed product development with integrated digital mock-up solutions. 2017, ftp://ftp.software.ibm.com/software/applications/plm/resources/05_PLM_001119_DMU_Whitepaper_LR_1.pdf. [Acessed 14 June 2017].
[6] Roj R. A comparison of three design tree based search algorithms for the detection of engineering parts constructed with catia v5 in large databases. J Comput Des Eng 2014;1(3):161–72.
[7] Iyer N, Jayanti S, Lou K, Kalyanaraman Y, Ramani K. Three-dimensional shape searching: state-of-the-art review and future trends. Comput Aided Des 2005;37(5):509–30.
[8] Tangelder JW, Veltkamp RC. A survey of content based 3D shape retrieval methods. Multimedia Tools Appl 2008;39(3):441–71.
[9] Hilaga M, Shinagawa Y, Kohmura T, Kunii TL. Topology matching for fully automatic similarity estimation of 3d shapes. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. New York, NY, USA: ACM; 2001, p. 203–12. http://dx.doi.org/10.1145/383259.383282.
[10] Osada R, Funkhouser T, Chazelle B, Dobkin D. Shape distributions. ACM Trans Graph 2002;21(4):807–32.
[11] Corney J, Rea H, Clark D, Pritchard J, Breaks M, MacLeod R. Coarse filters for shape matching. IEEE Comput Graph Appl 2002;22(3):65–74.
[12] Cardone A, Gupta SK, Karnik M. A survey of shape similarity assessment algorithms for product design and manufacturing applications. J Comput Inf Sci Eng 2003;3(2):109–18.
[13] Novotni M, Klein R. Shape retrieval using 3d zernike descriptors. Comput Aided Des 2004;36(11):1047–62.
[14] Hong T, Lee K, Kim S. Similarity comparison of mechanical parts to reuse existing designs. Comput Aided Des 2006;38(9):973–84.
[15] Biasotti S, Cerri A, Bronstein AM, Bronstein MM. Quantifying 3D shape similarity using maps: Recent trends, applications and perspectives. In: Eurographics (State of the Art Reports). 2014, p. 135–59.
[16] Cardone A, Gupta SK, Deshmukh A, Karnik M. Machining feature-based similarity assessment algorithms for prismatic machined parts. Comput Aided Des 2006;38(9):954–72.
[17] Li M, Zhang YF, Fuh JYH, Qiu ZM. Design reusability assessment for effective CAD model retrieval and reuse. Int J Comput Appl Technol 2011;40(1/2):3–12. http://dx.doi.org/10.1504/IJCAT.2011.038546.
[18] El-Mehalawi M, Miller RA. A database system of mechanical components based on geometric and topological similarity. part i: representation. Comput Aided Des 2003;35(1):83–94.
[19] El-Mehalawi M, Miller RA. A database system of mechanical components based on geometric and topological similarity. part ii: indexing, retrieval, matching, and similarity assessment. Comput Aided Des 2003;35(1):95–105.
[20] Chu C-H, Hsu Y-C. Similarity assessment of 3D mechanical components for design reuse. Robot Comput-Integr Manuf 2006;22(4):332–41.
[21] Çiçek A. Similarity and scaling assessments of mechanical parts using adjacency relation matrices. J Mater Process Technol 2008;206(1–3):106–19.
[22] Biasotti S, Marini S, Spagnuolo M, Falcidieno B. Sub-part correspondence by structural descriptors of 3D shapes. Comput Aided Des 2006;38(9):1002–19.
[23] Tao S, Huang Z, Ma L, Guo S, Wang S, Xie Y. Partial retrieval of cad models based on local surface region decomposition. Comput Aided Des 2013;45(11):1239–52.
[24] Zehtaban L, Elazhary O, Roller D. A framework for similarity recognition of CAD models. J Comput Des Eng 2016;3(3):274–85.
[25] Giannini F, Lupinetti K, Monti M. Identification of similar and complementary subparts in B-rep mechanical models. J Comput Inf Sci Eng 2017;17(4). 041004.
[26] Sacks E, Joskowicz L. The configuration space method for kinematic design of mechanisms. MIT Press; 2010.
[27] Li Z, Zhou X, Liu W. A geometric reasoning approach to hierarchical representation for B-rep model retrieval. Comput Aided Des 2015;62:190–202.
[28] Zheng Y, Cohen-Or D, Averkiou M, Mitra NJ. Recurring part arrangements in shape collections. Comput Graph Forum 2014;33:115–24.
[29] Su H, Maji S, Kalogerakis E, Learned-Miller E. Multi-view convolutional neural networks for 3d shape recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 945–953.
[30] Yi L, Guibas L, Hertzmann A, Kim VG, Su H, Yumer E. Learning hierarchical shape segmentation and labeling from online repositories, arXiv preprint arXiv:1705.01661.
[31] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: Deep learning on point sets for 3d classification and segmentation. Proc Comput Vis Pattern Recognit IEEE 2017;1(2):4.
[32] Huang Q, Koltun V, Guibas L. Joint shape segmentation with linear programming. ACM Tran Graph 2011;30:125.
[33] Paraboschi L, Biasotti S, Falcidieno B. 3D scene comparison using topological graphs. In: Eurographics Italian chapter conference; 2007. p. 87–93.
[34] Döllner G, Kellner P, Tegel O. Digital mock-up and rapid prototyping in automotive product development. J Integr Des Process Sci 2000;4(1):55–66.
[35] Riascos R, Levy L, Stjepandić J, Fröhlich A. Digital mock-up. In: Stjepandić J, Wognum N, Verhagen WJC, editors. Concurrent engineering in the 21st century: Foundations, developments and challenges. Springer International Publishing; 2015, p. 355–88. http://dx.doi.org/10.1007/978-3-319-13776-6_13.
[36] Chaudhuri S, Kalogerakis E, Guibas L, Koltun V. Probabilistic reasoning for assembly-based 3d modeling. ACM Trans Graph 2011;30:35.
[37] Wu Z, Song S, Khosla A, Yu F, Zhang L, Tang X, et al. 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 1912–20.
[38] Toche B, Huet G, McSorley G, Fortin C. A product lifecycle management framework to support the exchange of prototyping and testing information. In: ASME 2010 international design engineering technical conferences and computers and information in engineering conference; 2010. p. 1259–70.
[39] Shahwan A. Processing geometric models of assemblies to structure and enrich them with functional information [Ph.D. thesis], Grenoble; 2014.
[40] Swamidass PM. Encyclopedia of production and manufacturing management. Springer Science & Business Media; 2000.
[41] Whitney DE. Mechanical assemblies: Their design, manufacture, and role in product development, vol. 1. Oxford University Press on Demand; 2004.
[42] Turner JU, Subramaniam S, Gupta S. Constraint representation and reduction in assembly modeling and analysis. IEEE Trans Robot Autom 1992;8(6):741–50.
[43] Shahwan A, Foucault G, Léon J-C, Fine L. Towards automated identification of functional designations of components based on geometric analysis of a DMU. In: GTMG2011-12èmes Journées du Groupe de Travail en Modélisation Géométrique. 2011, p. 61.
[44] Shahwan A, Foucault G, Léon J-C, Fine L. Deriving functional properties of components from the analysis of digital mock-ups. Eng Comput 2014;16, URL https://hal.inria.fr/hal-00922958.
[45] Vilmart H, Léon J-C, Ulliana F. From cad assemblies toward knowledge-based assemblies using an intrinsic knowledge-based assembly model. Comput-Aid Des Appl 2018;15(3):300–17.
[46] Brière-Côté A, Rivest L, Maranzana R. Comparing 3D CAD models: Uses, methods, tools and perspectives. Comput-Aid Des Appl 2012;9(6):771–94.
[47] Deshmukh AS, Banerjee AG, Gupta SK, Sriram RD. Content-based assembly search: A step towards assembly reuse. Comput Aided Des 2008;40(2):244–61.
[48] Chen X, Gao S, Guo S, Bai J. A flexible assembly retrieval approach for model reuse. Comput Aided Des 2012;44(6):554–74.
[49] Pakkanen J, Huhtala P, Juuti T, Lehtonen T. Achieving benefits with design reuse in manufacturing industry. Procedia CIRP 2016;50:8–13.
[50] Renu RS, Mocko G. Computing similarity of text-based assembly processes for knowledge retrieval and reuse. J Manuf Syst 2016;39:101–10.
[51] Renu R, Mocko G. Retrieval of solid models based on assembly similarity. Comput-Aid Des Appl 2016;13(5):628–36.
[52] Katayama K, Sato T. Matching 3D CAD assembly models with different layouts of components using projections. IEICE Trans Inf Syst 2015;98(6):1247–50.

[53] Katayama K, Sato T. A matching method for 3D CAD models with different assembly structures using projections of weighted components. J Inf Process 2017;25:376–85.
[54] Wang P, Li Y, Zhang J, Yu J. An assembly retrieval approach based on shape distributions and Earth Mover's distance. Int J Adv Manuf Technol 2016;86(9–12):2635–51.
[55] Zhang J, Pang J, Yu J, Wang P. An efficient assembly retrieval method based on hausdorff distance. Robot Comput-Integr Manuf 2018;51:103–11.
[56] Hu K-M, Wang B, Yong J-H, Paul J-C. Relaxed lightweight assembly retrieval using vector space model. Comput Aided Des 2013;45(3):739–50.
[57] Tao S, Huang Z. Assembly model retrieval based on optimal matching. In: Software engineering and knowledge engineering: Theory and practice. Springer; 2012, p. 327–36.
[58] Miura T, Kanai S. 3D shape retrieval considering assembly structure. In: Proceeding of asian symposium for precision engineering and nanotechnology 2009; 2009. p. 11–3.
[59] Han Z, Mo R, Yang H, Hao L. CAD assembly model retrieval based on multi-source semantics information and weighted bipartite graph. Comput Ind 2018;96:54–65.
[60] Li Z, Zhou X, Liu W, Niu Q, Kong C. A similarity-based reuse system for injection mold design in automotive interior industry. Int J Adv Manuf Technol 2016;87(5–8):1783–95.
[61] Deshmukh AS, Gupta SK, Karnik MV, Sriram RD. A system for performing content-based searches on a database of mechanical assemblies. In: ASME 2005 international mechanical engineering congress and exposition. American Society of Mechanical Engineers; 2005, p. 411–23.
[62] Gupta SK, Cardone A, Deshmukh A. Content-based search techniques for searching CAD databases. Comput-Aided Des Appl 2006;3(6):811–9.
[63] Zhang J, Xu Z, Li Y, Jiang S, Wei N. Generic face adjacency graph for automatic common design structure discovery in assembly models. Comput Aided Des 2013;45(8–9):1138–51. http://dx.doi.org/10.1016/j.cad.2013.04.003.
[64] Wang P, Zhang J, Li Y, Yu J. Reuse-oriented common structure discovery in assembly models. J Mech Sci Technol 2017;31(1):297–307.
[65] Lupinetti K, Giannini F, Monti M, Pernot J-P. Multi-criteria retrieval of CAD assembly models. J Comput Des Eng 2018;5(1):41–53.
[66] Chen D-Y, Tian X-P, Shen Y-T, Ouhyoung M. On visual similarity based 3D model retrieval. Comput Graph Forum 2003;22:223–32.
[67] Ohbuchi R, Minamitani T, Takei T. Shape-similarity search of 3d models by using enhanced shape functions. Int J Comput Appl Technol 2005;23(2–4):70–85.
[68] Kazhdan M, Funkhouser T, Rusinkiewicz S. Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Symposium on geometry processing, vol. 6; 2003. p. 156–64.
[69] Renu RS. Product-process coupling to enable continuous improvement of assembly processes [Ph.D. thesis], Clemson University; 2016.
[70] Rubner Y, Tomasi C, Guibas LJ. The earth mover's distance as a metric for image retrieval. Int J Comput Vis 2000;40(2):99–121.
[71] Dubuisson M-P, Jain AK. A modified hausdorff distance for object matching. In: Pattern recognition, 1994 1-conference a: Computer vision & image processing., proceedings of the 12th IAPR international conference on, vol. 1. IEEE; 1994, p. 566–8.
[72] Kuhn HW. The hungarian method for the assignment problem. In: 50 Years of Integer Programming 1958-2008; 2010. p. 29–47.
[73] Munkres J. Algorithms for the assignment and transportation problems. J Soc Ind Appl Math 1957;5(1):32–8.
[74] Jonker R, Volgenant A. A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing 1987;38(4):325–40. http://dx.doi.org/10.1007/BF02278710.
[75] Gale D, Shapley LS. College admissions and the stability of marriage. Amer Math Monthly 1962;69(1):9–15.
[76] Miyazaki S, Iwama K. A survey of the stable marriage problem and its variants. In: 2008 international conference on informatics education and research for development of knowledge society infrastructure, vol. 00; 2008. p. 131–6. http://dx.doi.org/10.1109/ICKS.2008.7.
[77] Miller GA. Wordnet: a lexical database for english. Commun ACM 1995;38(11):39–41.
[78] Sánchez D, Batet M, Isern D, Valls A. Ontology-based semantic similarity: A new feature-based approach. Expert Syst Appl 2012;39(9):7718–28.
[79] Conte D, Foggia P, Sansone C, Vento M. Thirty years of graph matching in pattern recognition. Int J Pattern Recognit Artif Intell 2004;18(03):265–98.
[80] Cordella LP, Foggia P, Sansone C, Vento M. A (sub)graph isomorphism algorithm for matching large graphs. IEEE Trans Pattern Anal Mach Intell 2004;26(10):1367–72. http://dx.doi.org/10.1109/TPAMI.2004.75.
[81] Ip CY, Regli WC, Sieger L, Shokoufandeh A. Automated learning of model classifications. In: Proceedings of the eighth ACM symposium on solid modeling and applications. ACM; 2003, p. 322–7.
[82] Cardone A, Gupta S. Similarity assessment based on face alignment using attributed vectors. Comput-Aided Des Appl 2006;3(5):645–54.
[83] Ma L, Huang Z, Wang Y. Automatic discovery of common design structures in CAD models. Comput Graph 2010;34(5):545–55. http://dx.doi.org/10.1016/j.cag.2010.06.002.
[84] Lupinetti K, Chiang L, Giannini F, Monti M, Pernot J-P. Regular patterns of repeated elements in CAD assembly model retrieval. Comput-Aid Des Appl 2017;14(4):516–25.
[85] Rucco M, Giannini F, Lupinetti K, Monti M. A methodology for part classification with supervised machine learning. Artific Intell Eng Des Anal Manuf. 1–14. http://dx.doi.org/10.1017/S0890060418000197.
[86] Lupinetti K, Giannini F, Monti M, Pernot J-P. Identification of functional sets in mechanical assembly models. In: ICIDM2017 conference (Milan, Italy, 2017 July 17–19); 2017.
[87] Bunke H, Foggia P, Guidobaldi C, Sansone C, Vento M. A comparison of algorithms for maximum common subgraph on randomly connected graphs. In: Structural, syntactic, and statistical pattern recognition. Springer; 2002, p. 123–32.
[88] Pelillo M. Replicator equations, maximal cliques, and graph isomorphism. Neural Comput 1999;11(8):1933–55.
[89] Rucco M, Lupinetti K, Giannini F, Monti M, Pernot J-P. CAD assembly retrieval and browsing. In: Ríos J, Bernard A, Bouras A, Foufou S, editors. Product lifecycle management and the industry of the future. Cham: Springer International Publishing; 2017, p. 499–508.
[90] Msaaf Omar MR, Louis R. Part data mining for information re-use in a plm context. ASME 2007;187–94.
[91] Bai J, Gao S, Tang W, Liu Y, Guo S. Semantic-based partial retrieval of CAD models for design reuse. In: 2009 SIAM/ACM joint conference on geometric and physical modeling. ACM; 2009, p. 271–6.
[92] Yang RD, Fan X, Wu D, Yan J. Virtual assembly technologies based on constraint and dof analysis. Robot Comput-Integr Manuf 2007;23(4):447–56.
[93] Park SC, Oh JW. Kinetic model extraction from a geometric model. Comput-Aid Des Appl 2015;12(3):338–43.
[94] Lupinetti K, Giannini F, Monti M, Pernot J-P. Automatic extraction of assembly component relationships for assembly model retrieval. Procedia CIRP 2016;50:472–7.
[95] Swain AK, Sen D, Gurumoorthy B. Extended liaison as an interface between product and process model in assembly. Robot Comput-Integr Manuf 2014;30(5):527–45.
[96] Shahwan A, Léon J-C, Foucault G, Trlin M, Palombi O. Qualitative behavioral reasoning from components' interfaces to components' functions for DMU adaption to FE analyses. Comput Aided Des 2013;45(2):383–94.
[97] Ip CY, Regli WC. Content-based classification of CAD models with supervised learning. Comput-Aided Des Appl 2005;2(5):609–17.
[98] Pernot J-P, Giannini F, Petton C. Thin part identification for CAD model classification. Eng Comput 2015;32(1):62–85.
[99] Qin F-w, Li L-y, Gao S-m, Yang X-l, Chen X. A deep learning approach to the classification of 3D CAD models. J Zhejiang Univ Sci C 2014;15(2):91–106.
[100] Laga H, Mortara M, Spagnuolo M. Geometry and context for semantic correspondences and functionality recognition in man-made 3d shapes. ACM Trans Graph 2013;32(5):150.
[101] Jayanti S, Kalyanaraman Y, Iyer N, Ramani K. Developing an engineering shape benchmark for CAD models. Comput Aided Des 2006;38(9):939–53.
[102] Pu J, Lou K, Ramani K. A 2D sketch-based user interface for 3D CAD model retrieval. Comput-Aided Des Appl 2005;2(6):717–25.
[103] Pu J, Ramani K. A 3D model retrieval method using 2D freehand sketches. Comput Sci–ICCS 2005;2005:17–77.
[104] Eitz M, Hildebrand K, Boubekeur T, Alexa M. Sketch-based 3D shape retrieval. In: ACM SIGGRAPH 2010 Talks. ACM; 2010, p. 5.
[105] Liu Y-J, Luo X, Joneja A, Ma C-X, Fu X-L, Song D. User-adaptive sketch-based 3D CAD model retrieval. IEEE Trans Autom Sci Eng 2013;10(3):783–95.
[106] Tatsuma A, Koyanagi H, Aono M. A large-scale shape benchmark for 3D object retrieval: Toyohashi shape benchmark. In: Signal & information processing association annual summit and conference (APSIPA ASC), 2012 Asia-Pacific. IEEE; 2012, p. 1–10.
[107] Bespalov D, Ip CY, Regli WC, Shaffer J. Benchmarking CAD search techniques. In: Proceedings of the 2005 ACM symposium on solid and physical modeling. ACM; 2005, p. 275–86.