Retrieving Reusable 3D CAD Models Using Knowledge-Driven Dependency Graph Partitioning

Min Li[1], Y. F. Zhang[2] and J. Y. H. Fuh[3]

[1]National University of Singapore, limin@nus.edu.sg
[2]National University of Singapore, mpezyf@nus.edu.sg
[3]National University of Singapore, mpefuhyh@nus.edu.sg

ABSTRACT

3D design retrieval works, which focus on geometric representations, can hardly obtain reusable CAD models. We propose a novel method to retrieve reusable CAD models using knowledge-driven partitioning based on modeling dependency graphs. In order to retrieve reusable results, two dependency-graph partitionings are given based on design knowledge. The first is a horizontal partitioning scheme to simplify CAD models and preserve their essential shapes. In order to support partial shape retrieval and reuse, another vertical partitioning segments sub-parts in a meaningful way. The main contributions include (1) a graph-based modeling knowledge representation; (2) knowledge-driven graph partitioning strategies for improving reusability of retrieved results; and (3) redesign supports by utilizing modeling expertise. The validity of the method is demonstrated with several retrieval cases of realistic CAD models.

Keywords: 3D retrieval, design reuse, graph partitioning, partial similarity, CAD.
DOI: 10.3722/cadaps.2010.417-430

1 INTRODUCTION

A large proportion of computer-aided design (CAD) models can be reused to facilitate product development [29]. Text-based retrieval, which was used in PDM systems, are imprecise due to a lack of names of sufficient quality [20]. Therefore, retrieving models by content is more accurate [13, 28, 31].

But in design reuse contexts, similarity of geometric content is not always equal to mechanical reusability. 48% of the surveyed firms complained that parts found by content-based retrieval are inflexible to reuse [18]. The reason is that content-based 3D retrieval tools only work on geometric representations (such as mesh or surface models). However, reuse is a knowledge-intensive task that involves the analysis of the commonality between a reference and a target, as well as the feasibility to adapt the reference to the target. Since engineering knowledge is absent in content-based methods, it results in the less reusable retrieved results. This is a big gap between the retrieval and reuse.

In this paper, the aforementioned gap is bridged by our proposed CAD retrieval method, which takes advantage of the design knowledge in feature-based CAD models. The term *feature-based* refers to widespread use of the parametric and feature-based modeling technique. In CAD models, features are employed to capture modeling procedures, and convey high-level engineering semantics [27]. In the proposed method, the knowledge-based analysis of the feature interdependency is incorporated into the similarity assessment to enable the reuse-oriented CAD model retrieval. An attributed graph representation, namely feature dependency directed acyclic graph (FDAG), captures the feature interdependency. By analyzing the interdependency knowledge, we develop two FDAG partitioning schemes to extract feature-based components from CAD models: a horizontal FDAG partitioning (see Sec. 4.1) is to simplify fully detailed CAD models to less detailed; on the other hand, FDAG can be partitioned in a vertical way (see Sec. 4.2), and CAD models are segmented into sub-parts accordingly. A characteristic of the knowledge-driven partitioning is that it conforms to pre-defined modeling dependencies so that the resultant components are reusable in terms of the feature-based variant design. Therefore, reusability is maximally preserved.

A guide of the paper is given as follows. Sec. 2 investigates prior approaches in content-based 3D retrieval. In Sec. 3, the FDAG graph representation is presented. In the following section, two FDAG partitioning schemes (horizontal and vertical) are detailed to extract reusable CAD components for different reuse proposes. In Sec. 5, we formulate the essential and partial shape similarities of CAD models, and their retrieval algorithms. Tested cases of Sec. 6 illustrate that retrieved CAD components can be easily reused as their feature modeling knowledge is well preserved during the retrieval. In the meanwhile, best practices of original designs are inherently transferred to reused parts. Validity of the formulated algorithms is also tested with hundreds of real-world CAD models. Finally, the paper ends with conclusions.

2 OVERVIEW OF PRIOR APPROACHES

As an application of 3D graphics to the information retrieval (IR) problem, content-based 3D retrieval aims at retrieving 3D shapes by their actual contents [17]. There are two kinds of 3D similarities: global and partial shape similarity [32]. The former assesses how visually similar 3D objects are, while the latter is used to find a shape of which a part is similar to a part of another 3D object.

2.1 Global Shape Retrieval

A number of methods [16, 21, 22] assess global shape similarity by comparing statistical values of 3D objects. Shape distribution [22] is popular due to its efficiency and simplicity, but it only performs well for simple shapes. Another method is Surface Partitioning Spectrum (SPS) [25]. Other studies characterize 3D geometries by vector invariants, *e.g.* moments and spherical harmonics [26]. Although the above algorithms are efficient, a common drawback is trade-off between descriptor resolution and discrimination power. Moreover, they are less sensitive to details.

3D similarity can also be assessed by comparing 2D views [11, 24]. The rationale is that 3D shapes are equal if they look alike from any angles. Disadvantages of view-based methods are obvious: abandoned high-level 3D information, and inability to deal with internal spaces. Moreover, it is difficult to select ideal characteristic views [23].

The topology is another facet of 3D information. There are two ways to get 3D topology. The first is Reeb graph method [14], which splits 3D objects into parts and takes parts' connectivity as the topology. The second kind of methods represents 3D topology by skeletal graphs [12]. In CAD domain, structures of B-rep entities are compared in [10]. However, since graph matching is a NP-problem, a simple part may have a too complicated B-rep graph to match.

In engineering information retrieval, domain knowledge is vital. Machining features of prismatic parts have been compared to estimate manufacturing costs of new parts [7]. Moreover, the similarity were assessed by sub-graph isomorphism of machining feature graphs [9]. The above studies utilize manufacturing expertise. The Ref. [8, 15, 19] adopted recognized features for CAD model comparison. But feature recognition is ambiguous due to the multiple-interpretation issue. Moreover, recognized features cannot reason out how these models have been modeled. Modeling expertise of a part, including modeling dependency specified during design process, has a significant impact on the reuse of the part.

2.2 Partial Shape Retrieval

As opposed to a full object, similar portions are matched by partial similarity retrieval [31], in which 3D segmentation plays a key role to extract comparable parts, for example, fingers of a hand model.

One popular criterion of 3D segmentation is the surface smoothness. A boundary is detected if there is a sharp change of curvature [30]. Furthermore, partial shapes, in the form of segmented surfaces, were matched in a many-to-many manner [3]. Another group of methods applied the clustering technique on 3D volumes. 3D objects are either segmented by Reeb graphs in a parallel way [6] or evenly clustered in a symmetric way [4]. More recently, topological graphs and geometric data of partial shapes were compared in [1].

However, the aforementioned 3D segmentation methods are easy to be affected by minor changes of shapes being segmented. More importantly, these methods may produce segmentations that are meaningless to CAD model reuse, *e.g.* surface soups or shape fragments. Such poor results are "dump" surfaces or solids, which are of little value in design reuse. Furthermore, the direct reuse of any patch of freeform faces is still challenging.

3 KNOWLEDGE ACQUISITION AND REPRESENTATION

The proposed method assesses the mechanical reusability by incorporating modeling knowledge into CAD model retrieval. The knowledge incorporation enables the analysis of internal structures of CAD models. This section will introduce the acquisition and representation mechanisms of the required modeling knowledge.

3.1 Definition of Feature Dependency Relation

In the creation of a CAD model, features are added one by one, until the model is created. A feature is not added alone, and it may be built upon others. For example, an edge blend on the boundary of an extrusion depends on the extrusion. The feature dependency can be described as follows. Given two features f and g, if the creation of g is referred to f by geometric constraints [5], we say g depends on f. In other words, g is built after creating f due to the modeling reference to f. This binary relation is denoted as $f \rightarrow g$, where the arrow indicates the modeling precedence of f over g.

For a feature set F, following properties are true for features a, b, or c in F:

$$\begin{cases} \text{Irreflexivity}: \forall a \in F, \neg(a \rightarrow a) \\ \text{Asymmetry}: \forall a, b \in F, a \rightarrow b \Rightarrow \neg(b \rightarrow a) \\ \text{Transitivity}: \forall a, b, c \in F, a \rightarrow b \wedge b \rightarrow c \Rightarrow a \mapsto c \end{cases} \tag{3.1}$$

where $\mapsto$ stands for a transitive dependency.

The irreflexivity of the binary relation prevents a feature from depending on itself, while the asymmetry ensures that two features must not depend on each other. Moreover, the feature dependency is transitive due to the nature of the procedural feature modeling.

3.2 Graph-based Representation

A model created by a feature modeling system generally maintains two distinct representations. A geometric representation describes the shape of the CAD model in terms of B-rep entities, and another feature representation includes design features and their relationships. The proposed method acquires the required knowledge from the feature representation of CAD models.

The feature representation of a CAD model records the modeling process by a procedural design history. As the term *history* suggests, all feature constitutes present in a model are sorted chronologically in its design history. A chronological order is a *totally ordered set*, where any pair of feature constitutes are mutually comparable by their instantiation times. However, a predecessor feature in the design history is not necessary for a successor from the viewpoint of modeling precedence, as there are probably no geometric constraints between them.

Since the irreflexivity, asymmetry, and transitivity properties apply to the feature dependency, features of a model can be arranged in a *strictly partially ordered set*, where only a feature pair having the dependency relation is comparable. A strictly partially ordered set corresponds to a directed acyclic graph. In the proposed method, the feature modeling expertise, *e.g.* features and their interdependency are represented by a feature dependency directed acyclic graph (FDAG). In an FDAG, vertices are design features, and edge directions indicate the modeling precedence of the features. An FDAG representation of a CAD model *M* is constructed as follows:

1) The construction of vertices. For design features $\{f_1, f_2, \ldots, f_n\}$, vertices $\{v_{f1}, v_{f2}, \ldots, v_{fn}\}$ are put into an empty directed acyclic graph (called *G*) correspondingly.
2) The construction of edges. Traverse pairs of vertices of *G*. When a pair $<v_{fi}, v_{fj}>$ is visited, a directed edge is inserted from v_{fi} to v_{fj} if f_i depends on f_j directly (not transitively).

A part and its FDAG are shown in Fig. 1. Note edges indicate modeling precedence of features, which is a reverse order to the actual feature dependency. This notation aims to reason the feature interdependency. For example, two paths $F_1 \rightarrow F_3 \rightarrow F_5$ and $F_1 \rightarrow F_5$ clearly hint at that F_5 is a lower end of the modeling priority among $\{F_1, F_3, F_5\}$. If we organize FDAG nodes to put low-prioritized ones in the bottom as the figure shown, not only binary relations between features are captured by the FDAG, but it also depicts the whole modeling interdependency as a hierarchical structure.

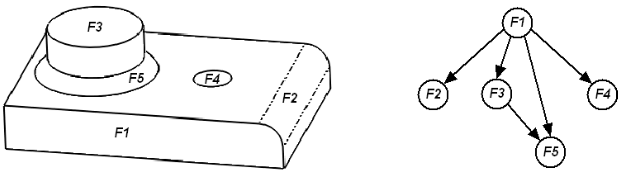


Fig. 1: A part and its FDAG representation.

4 KNOWLEDGE-DRIVEN GRAPH PARTITIONING

In design reuse, only few parts can be directly reused, while the vast majority must be adapted to the target shape. There are two ways to redesign a part: making minor modifications to a part or assembling of several sub-parts to fulfill requirements. Therefore, the search of reusable essential shapes and sub-parts is more important than finding out non-reusable models that are completely similar to queries. However, most 3D retrieval methods only compare mechanical CAD models as 3D rigid shapes. In the proposed method, CAD models are no longer assessed as rigid shapes with the help of modeling knowledge. In this section, the acquired knowledge (in the form of FDAG) is analyzed to develop appropriate FDAG graph partitioning schemes.

Two knowledge-driven FDAG partitioning schemes are developed in this section, to decompose complete models into essential shapes and sub-parts. The first in Sec. 4.1 is related a horizontal partitioning strategy, which corresponds to the essential shape simplification of CAD models. The second in Sec. 4.2 discusses a vertical FDAG partitioning scheme to segment meaningful sub-parts.

4.1 Horizontal FDAG Partitioning

A CAMI-ANC 101 part [27] and its FDAG diagram are shown in Fig. 2. A close observation at the figure reveals that minor features, *e.g.*, *ThurHoles*, *Pockets,* and *PinHoles* appear as *leaf* vertices, whose out-degrees are zero. The main reason of the small volume of these leaf nodes is that designers embody their designs in a coarse-to-fine way, and minor details are always built upon major components. At the same time, leaf nodes are the least significant ones in terms of the modeling priority. Based on these analyses, we can deduce the geometric and modeling significance of feature constitutes from the FDAG representation. A horizontal FDAG decomposition scheme is proposed as follows, to simplify CAD models from fully detailed to less detailed.

If $f \rightarrow g$ applies to features *f* and *g*, removing *f* leads to a deletion of *g* due to the modeling precedence, but not vice versa. That is, the *g* can be dropped without affecting *f* because of the asymmetry of the feature dependency. If we purge leaf nodes from an FDAG, the resultant CAD model is still valid in terms of modeling constraints after removing the corresponding features. In this way, details of a CAD model can be identified and simplified.

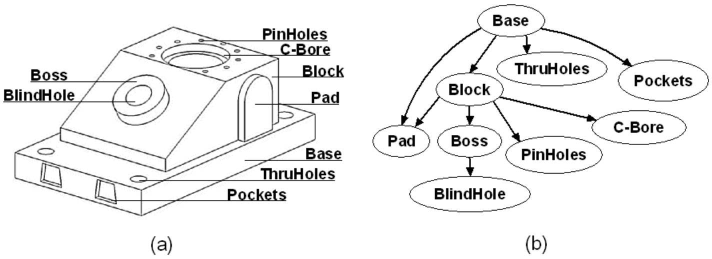


Fig. 2: Feature-based CAD model of the CAMI-ANC 101 part and its FDAG diagram.

4.1.1 *Essential Shape Simplification of CAD Models*

The basic idea of essential shape simplification algorithm is to peel leaf nodes from an FDAG layer by layer. By keeping the horizontal partitioning, a fully detailed CAD model is progressively simplified into less detailed ones. With details removed, essential shapes of CAD models can be obtained. The steps of the essential shape simplification algorithm are listed as follows:

1) Traverse vertices of the FDAG. For every vertex visited, mark it if its out-degree is zero. A nil out-degree indicates that the vertex is a leaf node and thus needs to be removed;
2) Traverse edges of the FDAG. Remove all edges pointing to the marked vertices;
3) Delete all marked vertices. Remaining sub-graph G_i corresponds to a simplified shape M_i of the full model M (i increments by 1 after every simplification).
4) The algorithm returns to the step 1) until a pre-defined termination criterion is reached.

The termination criterion is defined to prevent an original CAD model from being changed too much. The criterion is a threshold of the bounding box discrepancy between the original and a simplified shape. In our algorithm, bounding boxes are computed by the axis-aligned method. If $\{M_0, M_1, \ldots, M_i, \ldots, M_n\}$ represent a full CAD model and its n simplified shapes, $BB(M_i)$ indicates the bounding box of M_i, and $\Delta BB(M_i)$ stands for the bounding box discrepancy between M_0 and M_i (i = 0 to n). $|\Delta BB(M_i)| \leq \delta \times BB(M_0)$ should be satisfied. Based on our experiment, $\delta = 0.25$ works well in all cases.

Fig. 3 shows a pusher pad model and its essential shape simplification process, from fully detailed to less detailed, as indicated by M_i (i = 0 to 2). The simplification process is driven by the horizontal FDAG partitioning algorithm. In the partitioning, as shown in right bottom of the figure, leaf nodes are removed layer by layer. Every removal corresponds to a batch deletion of least important features, in terms of modeling priority and geometric significance.

4.2 Vertical FDAG partitioning

Sec. 4.1 introduced a horizontal partitioning algorithm to simplify CAD models; while this section proposes a vertical partitioning scheme to find out a reasonable FDAG sub-graph decomposition, which leads to a meaningful sub-part segmentation of the CAD model.

As mentioned in Sec. 3.2, feature constitutes captured by an FDAG are elements of a strictly partially ordered set, where only pairs having geometric constraint relations are comparable. In other words, some features are unrelated. Take an example of the pusher pad shown in Fig. 3, two vertically separated sub-graphs, {*F2*, *F4*, *F5*} and {*F6*, *F7*}, are independent to each other from the viewpoint of the modeling dependency. In other words, they can be revised independently. On the same time, these two sub-graphs correspond to geometrically cohesive sub-parts, which are also functionally complete from a mechanical perspective. {*F2*, *F4*, *F5*} is a rectangle bolt component, and {*F6*, *F7*} forms a counter-bored hole. Using the modeling independence deduced, the vertical partitioning of FDAG provides a way to segment CAD sub-parts.

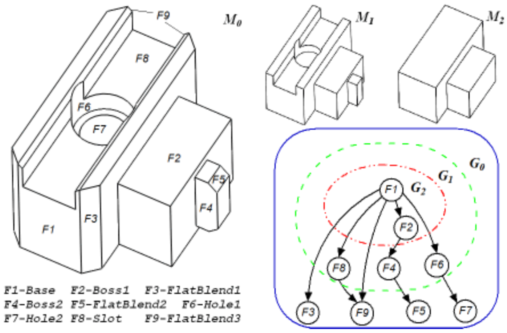


Fig. 3: A pusher pad model and its essential shape simplification process (M_0, M_1 and M_2), which is driven by the horizontal FDAG graph partitioning (G_0, G_1 and G_2).

4.2.1 Sub-part Segmentation of CAD Models

The key of the sub-part segmentation is to find out *articulation points* of an FDAG. A vertex is an articulation point (AP) if its removal renders the graph disconnected. In an FDAG, if we longitudinally decompose the FDAG at an AP, resultant sub-graphs are separated from the graph because only connection via the AP is broken. Correspondingly, a tentative sub-part is segmented from the CAD model.

Not all sub-parts segmented are meaningful. Meaningful sub-parts should be locally cohesive and globally decoupling: a cohesive sub-part has a connected shape; and a decoupling one has little dependency on other sub-parts. Here, a validation on tentatively segmented sub-parts is developed to choose meaningful ones. The sub-part segmentation algorithm consisting of the vertical FDAG partitioning and meaningful sub-part validation is given as follows.

1) Identify APs $\{ v_1, \ldots, v_i, \ldots, v_n \}$ of the FDAG G of a CAD model M. The variable i indicates the sequence of sub-graphs, ranging from 1 to the number of sub-graphs n. Traverse every identified AP:
   - For an AP v_i visited, compute the transitive adjacency closure of v_i as its tentative sub-graph. A transitive closure can be calculated by running depth-first search (DFS) on the v_i and marking all vertices in the DFS result as belonging to the closure.
2) Validate tentative sub-graphs $\{G_1, \ldots, G_i, \ldots, G_n\}$ by evaluating the reusability of sub-parts $\{P_1, \ldots, P_i, \ldots, P_n\}$. The $\{G_1, \ldots, G_i, \ldots, G_n\}$ were partitioned by APs $\{ v_1, \ldots, v_i, \ldots, v_n \}$, respectively. A valid sub-graph represents a sub-part that has mechanical meanings and yet easy to reuse.
   - If a tentative sub-graph G_i equals to the original FDAG G, it is not valid because the corresponding sub-part P_i is the complete model M;
   - If G_i introduces more incoming edges than those pointing to v_i, the sub-graph is not valid. The rationale is that more edges from the outside mean more modeling constraints for relocating the sub-part, which are the greatest barrier for reuse. Assume $D^{-}(\{v_i\})$ and $D^{-}(G_i)$ are the set of external dependencies of the AP v_i and sub-graph G_i respectively, the following equation must be hold:

$$D^{-}(G_i) \subseteq D^{-}(\{v_i\}) \tag{4.1}$$

     where two modeling dependencies are identical only if their constraint types and dependent entities are same. Please refer to [5] for a full list of geometric constraint types.
   - For a G_i, the geometric connectivity of P_i is checked for the sub-part cohesiveness.

We apply the sub-part segmentation algorithm on the pusher pad model of Fig. 3, and the segmentation results are shown in Fig. 4. By analyzing the FDAG, APs of the graph are automatically identified (circled by double lines) and tentative sub-graphs (circled by dash-dot lines) are decomposed by the proposed vertical partitioning. Among these tentative sub-graphs, a false result is shown in Fig.

4(b), which is dropped after the sub-graph validation. Three valid sub-graphs are listed in Fig. 4(c-e), corresponding sub-parts are shown as well. These sub-parts are geometrically connected CAD components that have limited dependencies to the rest the model so that the desirable cohesiveness and decoupling properties are satisfied. Especially, Fig. 4(e) shows a counterbore sub-part only consisting of negative features (*F8* and *F9* are suppressed to show the complete counterbore shape). Such subtractive CAD components are hard to be obtained by geometric-based segmentation algorithms.

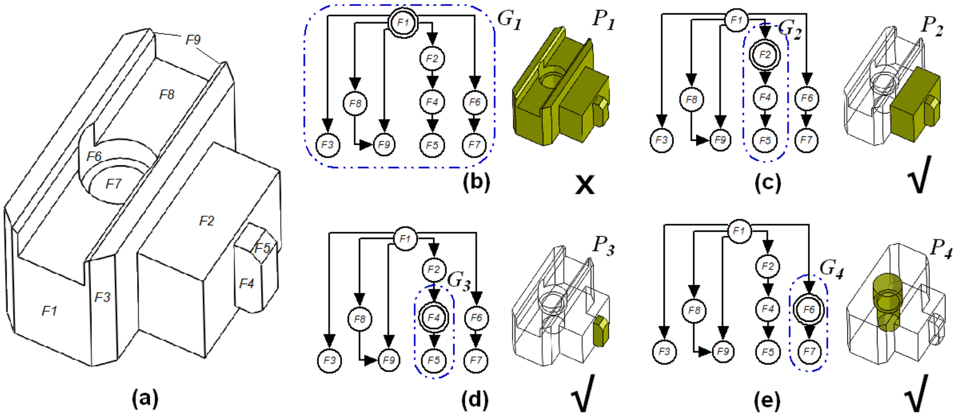


Fig. 4: (a) The pusher pad model of Fig. 3 (b-e) Tentative sub-graphs partitioned by FDAG articulation points (circled by double lines) and corresponding sub-parts. A false sub-part is (b), while (c-e) are valid sub-part segmentations. Especially, (e) shows a counterbore sub-part consisting of negative features solely.

5 REUSABLE CAD MODEL RETRIEVAL

Current 3D retrieval methods work on geometric representations (mesh or surface models), but these methods cannot obtain good results as they process models as rigid shapes. Rigid shapes are hard to reuse because redesigning a part needs expert understandings about how the part was built. The proposed method analyses a CAD model as a collection of interdependent feature constitutes so that extraction of reusable CAD components is possible. In Sec. 4, CAD models have been decomposed into two kinds of reusable components by knowledge-driven FDAG partitioning: the horizontal partitioning simplifies fully detailed CAD models into less detailed ones for the essential shape retrieval, and the vertical partitioning extracts cohesive and decoupling sub-parts for the partial shape retrieval. Sec. 5.1 and 5.2 will present the CAD model retrieval based on essential and partial shape similarity assessments, respectively.

5.1 CAD Model Retrieval by Assessing Essential Shape Similarity

The essential shape similarity assessment chooses archived CAD models that are similar to the query model in general shapes, while tolerate their differential details. In Sec. 4.1, details of full CAD models are simplified progressively. By comparing shapes without details, essentially similar models can be chosen, and details of the chosen parts can be easy to redesign with feature-based modifications. Prior to the similarity assessment, the simplification process of a model should be indexed by a concise representation, which is named the *essential shape aggregation* (ESA) descriptor. The ESA descriptor captures geometric and modeling information of all simplified shapes of the model.

Given a model M, its FDAG G and simplified essential shape M_i (i = 1 to n), every simplification M_i has a corresponding FDAG sub-graph G_i, which is created by the horizontal partitioning. The ESA descriptor of M is generated as follows:

1) Geometries of simplifications M_i (i = 1 to n) are characterized by SD D2 histogram [22] as the SD descriptor shows good discrimination power for shapes with less details [2].
   - Compute a D2 histogram H_i for a simplified shape M_i (i = 1 to n). The computation of the D2 histogram is similar to [22].
2) At the same time, FDAG sub-graphs of simplified shapes are kept as modeling information.
   - Generate the FDAG sub-graph G_i (i = 1 to n) in the form of adjacency lists.
3) Pairs of D2 histogram H_i and FDAG sub-graph G_i are aggregated into an ordered list, which is defined as the ESA descriptor of M.

Once archived models are indexed by ESA descriptors, users can query in a 3D form to search essentially similar CAD models. The essential similarity assessment process is defined as follows:

1) Compute an SD histogram H_Q for the given 3D query Q;
2) Compare ESA descriptors of archived CAD models with H_Q:
   - Given an archived model M with n simplifications, its ESA descriptor is $\{<H_1, G_1>, \ldots, <H_i, G_i>, \ldots, <H_n, G_n>\}$. We compare the distance between H_Q and H_i (i = 1 to n). We adopt the Manhattan distance as it outperforms other metrics in the SD histogram comparison [22]. A k-dimensional histogram is a vector in the space $\mathbf{R}^k$, and the Manhattan distance L_1 between two vectors $X <x_1, x_2, \ldots, x_k>$ and $Y <y_1, y_2, \ldots, y_k>$ as $L_1(X,Y) = \sum_{i=1}^{k} | x_i - y_i |$.
   - The shortest L_1 distance between H_i and H_Q (i = 1 to n) is the distance of M to Q. The corresponding M_i is the representative essential shape most similar to Q.

   $$L_1^{\min}(M, Q) = \min_{i \in [1,n]} L_1(H_i, H_Q) \tag{5.1}$$

   The *essential* shape *similarity* of M to Q can be defined as the following equation, where $L_1^{\max}$ is the theoretical maximum Manhattan distance between D2 histograms:

   $$Essential\_Shape\_Similarity(M, Q) = 1 - \frac{L_1^{\min}(M, Q)}{L_1^{\max}} \tag{5.2}$$
3) Sort all the archived CAD models by their similarities to Q in descending order. Top ranked models are retrieved to designers for reuse.

Retrieved results have similar essential shapes to the query. Moreover, as essential shapes are obtained by modeling dependency partitioning, differential details could be adjusted by certain feature-based modifications. At the same time, new features can be easily added because pre-defined geometric constraints and entities have been preserved during the retrieval.

5.2 CAD Model Retrieval by Assessing Partial Shape Similarity

The partial shape similarity assessment selects CAD models that only have portions similar to the query. By applying the vertical partitioning on the FDAG, cohesive and decoupling sub-parts are extracted as candidate partial shapes, which are further compared with the query to find the most similar one. A sub-part descriptor is developed to index each sub-part extracted. The generation of a sub-part descriptor is given as follows:

1) For a sub-part extracted from a CAD model M, says P:
   - Compute the SD D2 histogram H_P for P.
   - Calculate the adjacency list of the corresponding FDAG sub-graph G_P.
2) The pair of H_P and G_P is the sub-part descriptor.

Extracted sub-parts $\{P_1, P_2, \ldots, P_n\}$ are compared against a user-sketched query Q to find similar partial shapes. The partial shape similarity is assessed as follows:

1) Compute the similarity of the query Q to every sub-part, says P_i (i = 1 to n).

- Calculate the SD D2 histogram H_Q of the 3D query Q.
- Compute the Manhattan distance between H_Q and the histogram H_{Pi} of P_i. The distance is the dissimilarity of the query Q to the P_i.
- The *partial shape similarity* of Q to P_i is defined as follows. $L_1^{\max}$ is the maximum L_1 of D2.

$$Partial\_Shape\_Similarity(P_i, Q) = 1 - \frac{L_1(H_{Pi}, H_Q)}{L_1^{\max}} \tag{5.3}$$

2) Sort sub-parts from the most similar to the least similar and retrieve the top ranked to designers.

The top retrieved results are sub-parts most similar to the user-specified query. A sub-part is segmented by FDAG articulation points so that it only has limited dependencies to the rest of the model. Such a sub-part can be easily separated and relocated on a new target by resolving the limited dependencies. One more advantage is that the reused sub-part is fully constrained and well integrated with the target; therefore, it makes future retrieving of the reused sub-part possible.

6 IMPLEMENTATIONS AND EVALUATIONS

A prototype system is implemented by C++ and integrated with a CAD system (SolidWorks™) as plug-in. Users can finish new designs expeditiously by searching existing models. They can sketch what they want in a 3D way; the prototype plug-in captures the query and returns similar yet reusable CAD components; finally, the retrieved are reused in target designs with all modeling constraints preserved.

6.1 Evaluations on Essential Similarity

Fig. 5 illustrates a case study of matching a complex CAD model using a simple query, which is enabled by the proposed essential shape similarity assessment. The figure lists a pin-connector part M and some of its simplified shapes $\{M_4, M_7, M_{10}\}$. The progressive simplification is driven by the horizontal FDAG partitioning. Simplified shapes are characterized by an ESA descriptor, in which the geometric information is captured by a series of D2 histograms, *e.g.* $\{H_4, H_7, H_{10}\}$ in the figure. During the similarity assessment, the D2 histogram of the query Q is compared with every histogram in the ESA descriptor, and the *similarity* $(M_{10}, Q) = 0.96$ is selected as the essential shape resemblance of M to Q. The 0.96 is also the highest similarity of all archived CAD models to Q. Conversely, rigid shape matching methods without detail suppression cannot retrieve the M with such a simple query Q in early search. For instance, the SD D2 similarity between Q (i.e. H_Q) and M (i.e. H_0) is 0.83 only. Thus, users have to spend time in detailing the query to find out a desirable complex model.

We built a data set consisting of 627 realistic-scale CAD models. These models are manually classified into 32 categories. The models in a same category are *relevant* in terms of essential shapes.

The evaluation was made between the ESA and the SD [22] descriptors. Given a query model of Fig. 6(a), Fig. 6(b) listed five top ranked results by the ESA, which are all relevant; while the SD brought two irrelevant models (the 2nd and 4th in Fig. 6c). The irrelevancy can be explained that complex geometries of real-world CAD models negatively affect the SD descriptor, while the ESA copes with the complexity well. The precision-recall (P-R) curve is adopted for the quantitative comparison. P-R curves of ESA and SD descriptors were superimposed on Fig. 6, which indicates a better performance of ESA.

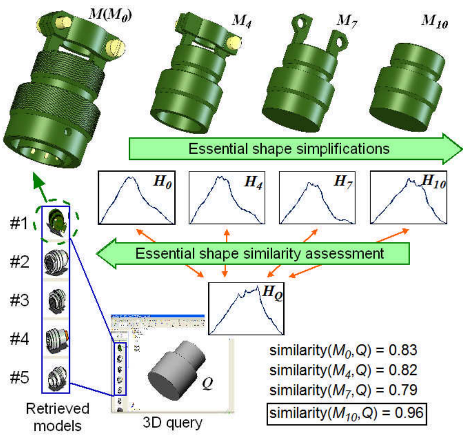


Fig. 5: Case study of real-world CAD model retrieval enabled by the proposed essential shape similarity assessment.

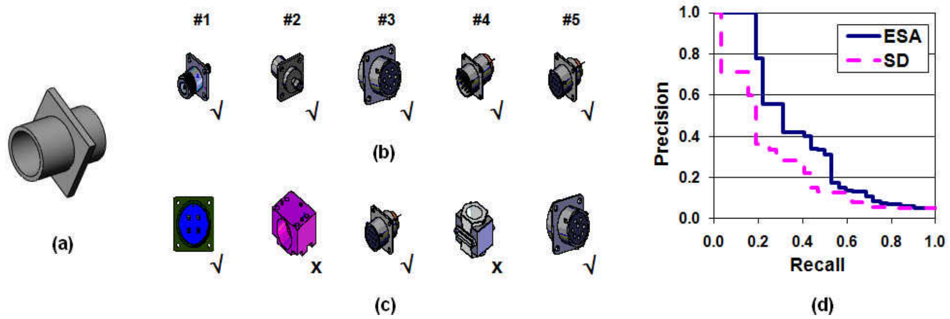


Fig. 6: (a) A query model; (b) and (c) High ranked retrievals by the essential shape aggregation (ESA) and shape distribution (SD) descriptors respectively; (d) Superimposed precision-recall curves (ESA: solid, SD: dashed).

The overall accuracy for all kinds of CAD models is benchmarked by an average P-R curve, the precision of which is the simple mean of precision rates of all plotted P-R curves. The simple precision mean is the sum of precisions at specified recall levels (*i.e.* ΣP^{λ}, where P^{λ} is the precision at the recall λ, and λ is a real number of [0, 1]) divided by the count of averaged curves m (in our experiment, one P-R curve is plotted for a model category, thus m is 32) as:

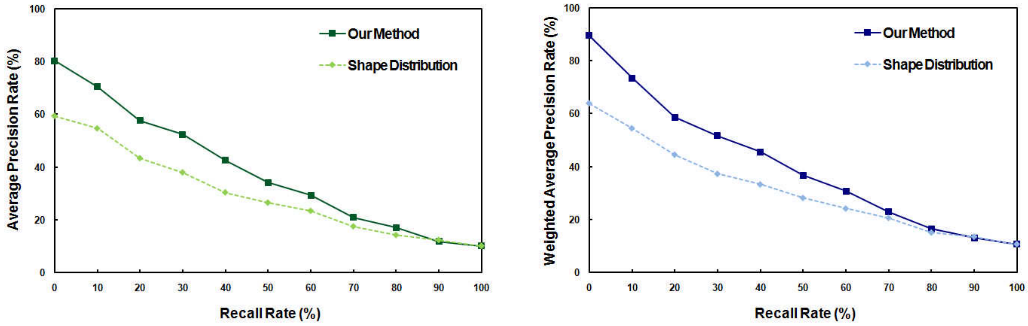
$$\overline{P} = \frac{1}{m}\sum_{i=1}^{m} P_i^{\lambda} \tag{6.1}$$

However, if model numbers of 32 categories are not balanced, the simple average may be distorted because a small category of few models has the equivalent proportion as large categories do. To eliminate the distortion, weights of categories are introduced into the average computation (see the following equation). The weight of a category is the model count of the category over the total number of CAD models evaluated.

$$\overline{P}_{\text{weighted}} = \frac{1}{m}\sum_{i=1}^{m} w_i P_i^{\lambda} \tag{6.2}$$

Fig. 7(a-b) showed simple and weighted average P-R curves of all the 32 categories, respectively. The higher curves of the ESA descriptor provides clear evidence that it performed better in retrieving broader model categories than the SD did.

Comparisons are further made in three recall ranges: 0 to 20%, 20 to 80%, and 80 to 100%, which correspond to high precision, middle recall, and high recall performance, respectively. Fig. 7b obviously reveals that the ESA descriptor was superior to the SD one in both high precision and middle recall ranges. When the recall was 10%, the ESA (73%) outperformed the SD (54%) by 19% in terms of retrieval precision. The outperformance confirms that in early search, the ESA descriptor was more efficient in relevant CAD model retrieval than rigid shape matching methods did. Discussions can be also made in the high recall range (0.8 to 1.0) as curves of ESA and SD were same. A possible explanation is that in this range, almost all relevant models have been retrieved; therefore, the precision, which is the proportion of the relevant items of the retrieved, is expected to flatten out if the remainder is purely irrelevant models.



(a) (b)

Fig. 7: (a) Average precision-recall (P-R) curves of the essential shape aggregation (ESA) and the shape distribution (SD) descriptors; (b) Weighted average P-R curves for the ESA and SD descriptors.

6.2 Evaluations on Partial Similarity

Fig. 8 demonstrates a case of retrieving similar CAD sub-parts by the proposed partial shape similarity assessment. Users can conveniently sketch a 3D query Q, which specifies the desirable partial shape, as shown in the middle-bottom of the figure. Since sub-parts have been segmented by the vertical FDAG partitioning, the partial shape similarity is assessed between Q and the segmented sub-parts to find out most similar ones. Highest ranked sub-parts $\{P_1, P_2, P_3, P_4\}$ (highlighted in yellow) were shown in the figure. The shown sub-parts closely resemble the Q as their similarities to Q are all larger than 0.97; however complete CAD models of these sub-parts are obviously different. Therefore, the case study demonstrates the ability of retrieving desirable partial shapes of the proposed CAD model retrieval method.

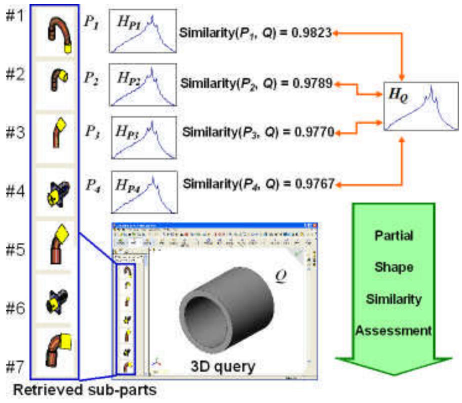


Fig. 8: Case study of CAD sub-part retrieval enabled by the proposed partial shape similarity assessment.

More examples of the CAD sub-part retrieval were shown in Fig. 9(a), where retrieved results on the right are apparently similar to the queries on the left. In addition, the bottom row showed a sub-part query specified by faces (in green) of an existing model. This example demonstrates that the proposed method provides flexible methods for composing 3D queries: either sketched3D manifolds or selected 3D surfaces are acceptable.

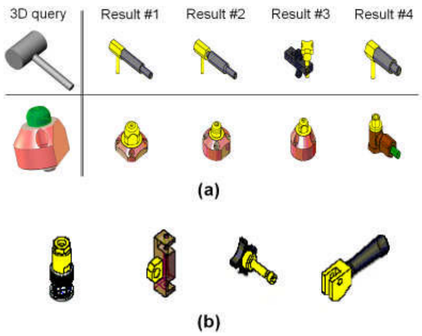


Fig. 9: (a) Sub-part queries and results (highlighted in yellow); (b) Extracted sub-parts (in yellow).

In our method, modeling expertise implied in CAD models is utilized to determine the sub-part segmentation. An advantage of this knowledge-based technique is that determining sub-parts is no longer affected by geometric factors, that is, the determination is invariable to minor changes of the shape being segmented, no matter whether surfaces have salient points, or how boundaries are surrounded by concavity. Moreover, the segmented sub-parts are feature constitutes that are mechanically meaningful. Some meaningful sub-parts were shown in Fig. 9(b).

Retrieved sub-parts can be reused easily in a feature-modeling system (SolidWorks in our case). Fig. 10 illustrated a sub-part reuse case. One sketches a query to search for desirable tapered pins (the top left screenshot). A retrieved sub-part is chosen as the reuse reference and the complete model containing the sub-part is loaded (the bottom left screenshot). As the FDAG sub-graph (the middle of the figure) shows that the retrieved sub-part depends to the rest of the model only by a locating dependency, the sub-part is externalized as a user-defined feature (UDF). On a new rectangular base, the UDF is relocated by re-configuring its locating dependency (the bottom right photo), and eventually the retrieved sub-part is reused and becomes a part of the fabric of the target design (the top right photo). In this way, the modeling knowledge of this sub-part is fully integrated in the new design so that this sub-part can be segmented, retrieved, and reused again in the future.

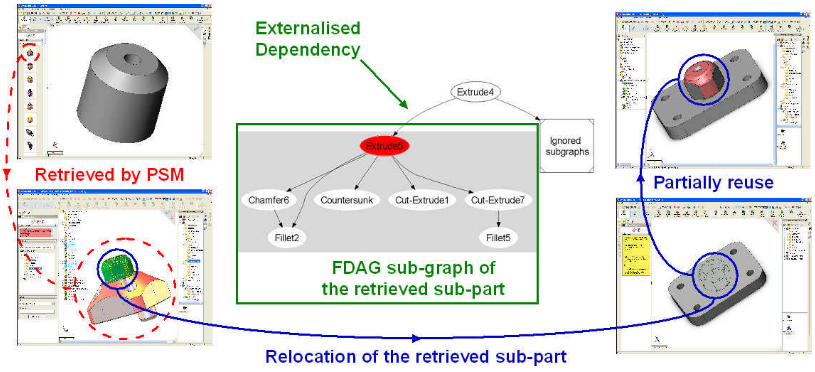


Fig. 10: Knowledge-preserving reuse of a tapered pin using the partial CAD model retrieval.

7 CONCLUSION

This paper proposes a knowledge-based method to retrieve reusable CAD models so as to bridge the current gap between CAD model retrieval and reuse. Specific contributions of the paper are described below:

- We have acquired feature modeling precedence knowledge from CAD models and developed a graph-based representation (FDAG) to capture the complicated modeling interdependency among feature constitutes.
- We have proposed two knowledge-driven FDAG partitioning schemes to extract reusable CAD components. With the partitioning, CAD models are no longer considered as rigid 3D shapes. Instead, using the horizontal FDAG partitioning, details of CAD models were progressively simplified to obtain essential shapes. On the other hand, sub-parts were extracted from full CAD models by vertical FDAG partitioning. The CAD component extractions conform to pre-defined modeling constraints. Therefore, extracted components are highly reusable in terms of variant and adaptive design.
- With the simplified essential shapes and segmented sub-parts, comparing the essential and partial shape similarity is possible. We have formulized these similarity measures and proposed CAD model retrieval algorithms based on the essential and partial shape similarity.
- We have successfully implemented a prototype system to demonstrate the feasibility of the proposed algorithms. We have also evaluated the effectiveness of the knowledge-based retrieval method on over 600 real CAD models. The results showed that the proposed method outperforms other 3D retrieval methods.

We expect that the proposed method will serve three purposes. First, it will allow designers to retrieve CAD models by specifying a desirable essential or partial shape. In this way, designers can retrieve essential similar parts (*e.g.* a part family) or meaningful sub-parts as redesign references. Second, it offers ease of reuse to designers as the modeling dependency analysis is incorporated into the CAD model retrieval to ensure the design reusability of the retrieved. Hence, design reuse inflexibility can be effectively prevented. Third, it will provide designers the access to original design expertise embedded in CAD models by visualizing FDAG graphs during the design reuse activities. Thus, best practices are inherently transferred to new designs.

REFERENCES

[1] Bai, J.; Liu, Y.; Gao, S.: 2007, Multi-mode solid model retrieval based on partial matching, Proceedings of 10th IEEE International Conference on Computer Aided Design and Computer Graphics, CAD/Graphics 2007, Beijing, China, October 15-18, 126-131.

[2] Bespalov, D.; Ip, C. Y.; Regli, W. C.; Shaffer, J.: 2005, Benchmarking CAD search techniques, Proceedings of the 2005 ACM symposium on Solid and physical modeling, Cambridge, Massachusetts, United States, June 13-15, 275–286.

[3] Bespalov, D.; Regli, W. C.; Shokoufandeh, A.: Local feature extraction and matching partial objects, Computer-Aided Design, 38(9), 2006, 1020–1037.

[4] Bespalov, D.; Shokoufandeh, A.; Regli, W. C.; Sun, W.: Scale-space representation and classification of 3D models, J. of Computing and Information Science in Engineering, 3(4), 2003, 315–324.

[5] Bettig, B.; Shah, J.: Derivation of a standard set of geometric constraints for parametric modeling and data exchange, Computer Aided Design, 33(1), 2001, 17-33.

[6] Biasotti, S.; Marini, S.; Spagnuolo, M.; Falcidieno, B.: Sub-part correspondence by structural descriptors of 3D shapes, Computer-Aided Design, 38(9), 2006, 1002–1019.

[7] Cardone, A.; Gupta, S. K.; Deshmukh, A.; Karnik, M.: Machining feature-based similarity assessment algorithms for prismatic machined parts, Comp.-Aided Design, 38(9), 2006, 954–972.

[8] Chu, C.-H.; Hsu, Y. C.: Similarity assessment of 3D mechanical components for design reuse, Robotics and Computer-Integrated Manufacturing, 22(4), 2006, 332–341.

[9] Cicirello, V. A.; Regli, W. C.: An approach to a feature-based comparison of solid models of machined parts, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 16(5), 2002, 385-399.

[10] El-Mehalawi, M.; Miller, R. A.: A database system of mechanical components based on geometric and topological similarity. Part ii: Indexing, retrieval, matching, and similarity assessment, Computer-Aided Design, 35(1), 2003, 95–105.
[11] Funkhouser, T.; Min, P.; Kazhdan, M.; Chen, J.; Halderman, A.; Dobkin, D.; Jacobs, D.: A search engine for 3D models, ACM Transactions on Graphics, 22(1), 2003, 83–105.
[12] Gao, W.; Gao, S. M.; Liu, Y. S.; Bai, J.; Hu, B. K.: Multiresolutional similarity assessment and retrieval of solid models based on dbms, Computer-Aided Design, 38(9), 2006, 985–1001.
[13] Gupta, S. K.; Cardone, A.; Deshmukh, A.: Content-based search techniques for searching CAD databases, Computer-Aided Design and Applications, 3(6), 2006, 811-819.
[14] Hilaga, M.; Shinagawa, Y.; Kohmura, T.; Kunii, T. L.: 2001, Topology matching for fully automatic similarity estimation of 3D shapes, Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, Los Angeles, United States, pp. 203-212.
[15] Hong, T.; Lee, K.; Kim, S.: Similarity comparison of mechanical parts to reuse existing designs, Computer-Aided Design, 38(9), 2006, 973–984.
[16] Ip, C. Y.; Lapadat, D.; Sieger, L.; Regli, W. C.: 2002, Using shape distributions to compare solid models, Proceedings of the seventh ACM symposium on Solid modeling and applications, Saarbrucken, Germany, June 17-21, pp. 273–280.
[17] Iyer, N.; Jayanti, S.; Lou, K.; Kalyanaraman, Y.; Ramani, K.: Three-dimensional shape searching: State-of-the-art review and future trends, Computer-Aided Design, 37(5), 2005, 509–530.
[18] Jackson, C.; Buxton, M.: The design reuse benchmark report - seizing the opportunity to shorten product development, Aberdeen Group, 2007, <http://www.aberdeen.com/summary/report/benchmark/DesRes_Out_3908.asp>.
[19] Mcwherter, D.; Peabody, M.; Regli, W. C.; Shokoufandeh, A.: Solid model databases: Techniques and empirical results, J. of Comp. and Information Science in Engineering, 1(4), 2001, 300-310.
[20] Min, P.; Kazhdan, M.; Funkhouser, T.: 2004, A comparison of text and shape matching for retrieval of online 3D models, Proceedings of 8th European Conference on Research and Advanced Technology for Digital Libraries, ECDL 2004. , Bath, UK, Vol. LNCS 3232, pp. 209–220.
[21] Ohbuchi, R.; Minamitani, T.; Takei, T.: Shape-similarity search of 3D models by using enhanced shape functions, Int. Journal of Computer Applications in Technology, 23(2-4), 2005, 70-85.
[22] Osada, R.; Funkhouser, T.; Chazelle, B.; Dobkin, D.: Shape distributions, ACM Transactions on Graphics, 21(4), 2002, 807-832.
[23] Paquet, E.; Rioux, M.; Murching, A.; Naveen, T.; Tabatabai, A.: Description of shape information for 2-d and 3-d objects, Signal Processing: Image Communication, 16(1), 2000, 103–122.
[24] Pu, J.; Ramani, K.: On visual similarity based 2d drawing retrieval, Computer-Aided Design, 38(3), 2006, 249–259.
[25] Rea, H. J.; Corney, J. R.; Clark, D. E. R.; Taylor, N. K.: 2004, A surface partitioning spectrum (sps) for retrieval and indexing of 3D CAD models, Proceedings of 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, 3DPVT 2004, Thessaloniki, Greece, September 6-9, 167–174.
[26] Saupe, D.; Vranil, D.: 3D model retrieval with spherical harmonics and moments, Pattern recognition, Vol. 2191/2001, Springer, 2001, 392-397.
[27] Shah, J. J.; Mäntylä, M.: Parametric and feature-based CAD/CAM: Concepts, techniques, and applications, John Wiley & Sons, New York, 1995.
[28] Shum, H.-Y.; Hebert, M.; Ikeuchi, K.: 1996, On 3D shape similarity, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 526-531.
[29] Sivaloganathan, S.; Shahin, T.: Design reuse: An overview, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 213(7), 1999, 641-654.
[30] Suzuki, M. T.: A dynamic programming approach to search similar portions of 3D models, The World Scientific Engineering Academy and Society Transaction on Systems, 3(1), 2004, 129-135.
[31] Tangelder, J. W. H.; Veltkamp, R. C.: A survey of content based 3D shape retrieval methods, Multimedia Tools and Applications, 39(3), 2008, 441-471.
[32] Veltkamp, R.; Latecki, L.: Properties and performance of shape similarity measures, Data science and classification, Springer Berlin Heidelberg, 2006, 47-56.