www.cadanda.com Computer-Aided Design AND APPLICATIONS

General and Partial Retrieval of CAD Models Based on Surface Region Partition

Songqiao Tao[1,2]

[1]Wuhan University of Technology, taosongqiao@163.com
[2]Wuhan Technical College of Communications, taosongqiao@163.com

ABSTRACT

In order to improve the retrieval effect and efficiency for complex models, a general and partial retrieval approach for CAD models based on surface region partition is presented in this paper. First, CAD models are transformed to face attributed relational graphs. Second, the surface boundary of a solid model is decomposed into local convex, concave and planar regions with the smallest amount by considering its salient geometric features. Then, a kind of region codes is adopted to describe the surface regions and their links for CAD models. Finally, the similarity between the two models is evaluated by the comparison of their region attributes codes. Experimental results have shown that this approach is able to support general retrieval and partial retrieval and their efficiency can meet the requirement of practical applications.

Keywords: surface region partition, general retrieval, partial retrieval, face attributed relational graph

1. INTRODUCTION

As many CAD models are already available from public and proprietary libraries, reuse of these models and the knowledge embedded is becoming an important way to facilitate new designs [1]. In order to search for the reference schemes, users spent more than 60% of design time on collecting and reading the related documents. Obviously, three-dimensional (3D) retrieval can help users efficiently locate the desired models for reuse. Consequently, since recently, 3D retrieval for CAD models has become an active research topic in the academic community.

Since the direct comparison for 3D models is not convenient, 3D descriptors generated from original models are adopted for the comparison purpose. 3D descriptors have been widely studied in computer graphics area and some results can be directly applied to CAD models. In computer graphics community, most 3D descriptors can be classified into histogram-based, transform-based, view-based, graph-based and the combinations of the above. Osada et al. [2] proposed a typical histogram-based approach called shape distributions, which transform 3D shape matching to the comparison of probability distributions. The sampled distributions are the distances between two points on model surface or angles between their normal vectors. The transform-based approaches include spherical harmonic and 3D Zernike, which represent 3D shape with the moments of the 3D object's volumetric function [3,4]. The moments are invariant to rotation, translation and scaling. View-based approaches adopt 2D projections of 3D models in different directions to describe 3D models and most shape descriptors for 2D images can be used to represent the shapes of the projections [5,6]. Graph-based methods include skeleton graphs [7], Reeb graphs [8], feature graphs [9] and face graphs [10], which can represent models from rough to exact. However, graph matching requires intricate matching algorithms.

For CAD models retrieval, El-Mehalawi et al. [11] index and retrieve 3D models based on face attributed relational graphs whose nodes and links respectively correspond to faces and edges. Li et al. [12] used feature dependency directed acyclic graphs (FDAGs) to represent CAD models and their decomposed components. FDAGs can capture some related engineering knowledge besides model shapes. Bai et al. [13] used the extended feature trees to capture CAD models' design feature. In order to improve retrieval efficiency, they created bitmap indices for each reusable subpart in the library as well as a query model. Biasotti et al. [14] adopted the extended Reeb graphs to describe the structural information of 3D models.

Taylor & Francis Taylor & Francis Group

The extended Reeb graphs usually have a reasonable size but the cost for graph generation is usually rather high. Bespalov et al. [15] proposed a graph approach to represent 3D models with extracted local features and their adjacency relationships. The feature-adjacency graphs may be small in size, but the local feature extraction based on scale–space decomposition is complicated. Saber et al. [16] introduced a graph method to describe 2D shapes with feature points on boundary and their distances. The feature-point graphs are easy to construct, but the graph vertex number may become too large for 3D CAD models. For the comparison of retrieval algorithms, Bespalov et al [17] have developed a classified CAD model database called the National Design Repository, which provides models in B-rep format.

Compared with general retrieval, partial retrieval needs much more computation because it has to determine which part of a model to be matched with query model. One way to improve the efficiency is to reduce the size of the problem by making the granularity of model elements larger. The faces like polygonal planes and cylindrical surfaces in B-rep models are not large enough when models are complex. Consequently, we need surface region partition, which divides models into new model elements with some basic characteristics, replacing faces in B-reps

In this paper, we use surface region partition to reduce the models' complexity. First, query model and library models are respectively divided into local convex, concave and planar regions with the minimal number. Then, we use region property codes to describe the surface regions and their links. Finally, the similarity between query model and library models is evaluated by the comparison of their region property codes. Experimental results have shown that this approach is able to support general retrieval and partial retrieval and their efficiency can meet the requirement of practical applications.

The rest of this paper is organized as follows. After the related terminology definitions and problem statements are presented in section 2, the surface region partition for CAD models is introduced in section 3. In section 4, a kind of region codes that describe the surface regions and their links are addressed in detail. Following this, general and partial retrieval for CAD models are respectively introduced in Section 5 and Section 6. Section 7 gives some results of the experiments for the proposed approach. Finally, the paper ends up with some conclusions in Section 8.

2. TERMINOLOGY DEFINITIONS AND PROBLEM STATEMENTS

2.1. Terminology Definitions

Since B-rep models can be conveniently exported into a model file with the STEP standard format, surface region partition for CAD models is developed based on the B-rep model in this paper. First, we transform CAD models to face attributed relational graphs (*FARG*s). The *FARG* is an ordered pair $G = (V, E)$, where V is its node set and E is its link set, respectively corresponding to model's faces and model's edges. Then, the boundary of a solid CAD model is decomposed into a set of local convex, concave and planar regions with the minimal number. Before introducing the approach for surface region partition, some basic concepts are first given below.

Similar to the approaches developed in [18] and [19], the surfaces of CAD models can be divided into convex surfaces, concave surfaces, planar surfaces and saddle surfaces while the edges are classified into convex edges, concave edges, tangent edges, convex-tangent edges and concave-tangent edges based on their external edge angle. In this text, saddle surfaces are not considered because they appear less in the mechanical parts.

Definition 1. *Induced Graph* (*IG*): For *FARG* $G = (\boldsymbol{V}, \boldsymbol{E})$ and $G^1 = (\boldsymbol{V}^1, \boldsymbol{E}^1)$, G^1 is defined as *G's IG,* if they satisfy the following two conditions:

(1) $G^1.V^1 \subseteq G.V$;
(2) $G^1.E^1 = \{e | \text{for all } e \in G.E,\ e.v_i \in G^1.V^1, i = 1, 2\}$.

Definition 2. *Region type:* For a given region, we define it as *Planar Region* (*PR*) if all its surfaces are planar type while each interior edge is tangent type. If this region isn't a *PR* and all its surfaces are planar or convex (concave) type and each of interior edges are convex (concave), convex (concave) - tangent, or tangent type, we call it as *Convex Region* (C_vR) (*Concave Region* (C_cR)).

Corresponding to *PR*, C_VR and C_cR, *IG* can be classified into *Planar Region Graph* (*PRG*), *Convex Region Graph* (C_VR) and *Concave Region Graph* (C_cRG).

Definition 3. *Surface region partition*: Let $\boldsymbol{S} = \{G^1, G^2, \ldots, G^n\}$ be *IG* set, when a *FARG* $G = (\boldsymbol{V}, \boldsymbol{E})$ is decomposed into $\boldsymbol{S}$ and it satisfies the following conditions:

(1) $\boldsymbol{G}.V = G^1.V^1 \cup G^2.V^2 \ldots G^n.V^n$;
(2) $G^i.V^i \cap G^j.V^j = \emptyset$ if $i \neq j$.

Where G^i is a *PRG*, C_VRG or C_cRG $(i = 1, 2, \ldots, n)$, then $\boldsymbol{S}$ is called G's surface region partition.

2.2. Problem Statements

For a given *FARG* $G = (\boldsymbol{V}, \boldsymbol{E})$, the surface region partition is just the following optimization problem:

Find $\boldsymbol{S} = \{G^1, G^2, \ldots, G^n\}$;
Minimize $|\boldsymbol{S}|$;
Subject to: $\boldsymbol{S}$ is a surface region partition of $\boldsymbol{G}$.

3. SURFACE REGION PARTITION

Since *FARG* provides complete face attributed relational information, surface merging seems to be a natural approach. However, it may not be effective and efficient enough for CAD models. In order to balance the effect and the efficiency, surface region partition is accomplished by two steps (see Fig. 1) in this paper. First, CAD model is decomposed into the set $\boldsymbol{S}^{c} = \{G^1, G^2, \ldots, G^i, \ldots, G^n\}$ by performing initial partition. Here G^i may be *PRG*, $C_V RG$ or $C_C RG$. Then, we utilize optimization procedure to turn the initial $\boldsymbol{S}^{c}$ into the optimal result $\boldsymbol{S}$. In Fig. 1, *FARG G* is firstly decomposed into $\boldsymbol{S}^{c} = \{G^1, G^2, G^3\}$. Then, the optimization procedure merges G^2 and G^3 into one $C_C RG$ G^4. So, the final partition is $S = \{G^1, G^4\}$.

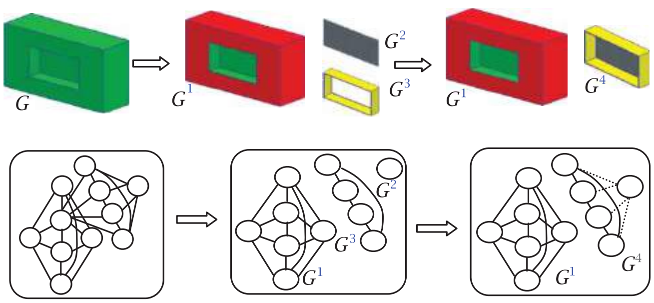


Fig. 1: An example for surface region partition: (a) FARG G, (b) Initial decomposition result S^c, (c) Optimized result S.

3.1. Initial Partition

In initial partition, we need to recognize the faces that definitely belong to a *PR*, $C_V R$ or $C_C R$. In order to explain these theories exactly, we firstly give the vertices definition for a *FARG G* in the following.

DEFINITION 4. *Vertex types*: Let $\boldsymbol{E}_i$ be the edges set of node $G.v_i$ in a *FARG G*, if $G.v_i$ represents convex(concave) or planar type and $\boldsymbol{E}_i$ isn't concave(convex) edge, $G.v_i$ is called *Convex* (*Concave*) *vertex* which is denoted by using symbol '+'("−"). Otherwise, it is a *Hybrid vertex* of G, and it is denoted using symbol '/'.

In above definition, we can find that there is a little bit difference between face convexity and the vertex types. Here, a planar may be called a convex (concave) vertex if its edges have consistent convex (concave) convexity. From the definition of vertex types, we can conclude that convex or concave vertex $G.v_i$ should have the same vertex type in G and its subgraph. But for hybrid vertex, $G.v_i$ may be convex or concave vertex in G's subgraph. The reason is that $G.v_i$'s partial adjacency edges don't exist in G's subgraph. In Fig. 2, "+", "−"and "/" respectively describe convex edge, concave edge and tangent edge. In Fig. 2(a) and 2(c), vertex v_1 belongs to convex type. But for hybrid vertex v_2, it is concave type in G's subgraph G^1 while it is convex type in G's subgraph G^2.

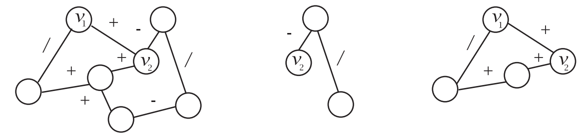


Fig. 2: An example for vertex type: (a) *FARG G*, (b) G's subgraph G^1, (c) G's subgraph G^2.

Based on above analysis, we can implement initial partition. First, we use **Procedure 1** to find $C_V RG$s or $C_C RG$s from *FARG G*directly by recognizing the maximal connected subgraphs with convex vertices or concave vertices. The reminder subgraphs with hybrid vertices are turned into convex or concave vertices by deleting their connected concave or convex edges in **Procedure 2**. **Procedure 1** and **Procedure 2** are given below.

PROCEDURE 1. ***$C_V RG$($C_C RG$) recognition algorithm*** (input G, output S_1^+, S_1^-, S^h):

Step 1 Set $S_1^+ = \emptyset, S_1^- = \emptyset, S^h = \emptyset$;
Step 2 Find all G's hybrid vertices, delete their connected edges;
Step 3 Delete concave (convex) edges of convex (concave) vertices in G;
Step 4 Recognize G's maximal connected subgraphs $S_1^+ = \{G_1^+, G_2^+, \ldots, G_i^+, \ldots, G_m^+\}$ ($S_1^- = \{G_1^-, G_2^-, \ldots, G_j^-, \ldots, G_n^-\}$) with convex (concave) vertices and convex (concave) edges and delete them from G;
Step 5 Check whether G_i^+ (G_j^-) in G is itself. If not, remove it from S_1^+ (S_1^-) and put it to G;
Step 6 If there is only one vertex in $G_i^+(G_j^-)$, remove it from S_1^+ (S_1^-) and put it to G;
Step 7 For the remainder subgraphs in G, recover the deleted edges and put them to $\boldsymbol{S^h}$.
Step 8 Output S_1^+, S_1^-, S^h.

PROCEDURE 2. *Hybrid* ***subgraph recognition algorithm*** (input $\boldsymbol{S^h}$, output S_2^+, S_2^-, S^c):

Step 1 Set $S_2^+ = \emptyset, S_2^- = \emptyset, S^c = \emptyset; S_1 = \emptyset, S_2 = \emptyset$;
Step 2 Remove all the convex edges in $\boldsymbol{S^h}$ and regenerate their vertex types;
Step 3 Call ***$C_C RG$ recognition algorithm*** (S^h, S_2^+, S_2^-, S_1);
Step 4 Remove all the concave edges in S_1 and regenerate their vertex types;
Step 5 Call ***$C_V RG$ recognition algorithm*** (S_1, S_2^+, S_2^-, S_2);
Step 6 If both S_2^+ and S_2^- are empty, generate a single-vertex subgraph for each vertex of S_2, and put them to S^c; exit the procedure.

Step 7 If $S^h \neq \emptyset$, go to Step 2; else, exit the procedure;
Step 8 Output S_2^+, S_2^-, S^c.

The above procedure is an iterative process and each iteration can find a maximum adjacency subgraph $C_V RG$ and $C_C RG$.

3.2. Optimization Initial Partition

After initial partition, *FARG G* is decomposed into $S^c = S_1^+ \cup S_1^- \cup S_2^+ \cup S_2^- \cup S^0$. However, the initial partition produces more surface regions than expected. For example, they may generate a $C_C RG$ and a *PRG* but actually the two regions should be regarded as a $C_C RG$ region. Then, we adopt an optimal procedure to handle this issue. In this process, there are two issues that need to be addressed. The first is that the mergeable condition and the second is whether a merging can improve the current partition. The mergeability of two subgraphs is that the induced graph from all the vertices of them is a $C_V RG$, $C_C RG$ or *PRG*. This means that the two subgraphs G^1 and G^2 have the consistent convexity while all edges existing between vertices of the two subgraphs have the consistent convexity with the two subgraphs, too. The improvement condition for region merging is that the optimized result has the smallest amount of surface region. Because the two mergeable subgraphs should be adjacent, we use the maximal mergeable adjacency subgraphs to meet this requirement in this paper. Consequently, we give the initial partition optimization algorithm in the following.

PROCEDURE 3. *Initial partition optimization algorithm* (input $\boldsymbol{S}^c$, output $\boldsymbol{S}$):

Step 1 Set $S^c = \{G^1, G^2, \ldots, G^n\}$, the merging flag *a*=false, $S = \emptyset$;
Step 2 For each surface region G^i and its adjacency subgraphs in S^c, if G^i's maximal mergeable subgraphs G^{imax} can be found, merge them and set *a*=ture;
 Step 2.1 If $a =$ ture, remove G^i and its merged adjacency subgraphs from S^c and put G^{imax} to $\boldsymbol{S}$;
 Step 2.2 If $a =$ false, remove G^i from S^c and put it to $\boldsymbol{S}$;
 Step 2.3 Go to the next subgraph.
Step 3 If $S^c = \emptyset$, return S as the final result of surface region *partition* and exit; otherwise, update $S^c = \{G^1, G^2, \ldots, G^n\}$ and go to Step 2

3.3. The Efficiency and Effect for Surface Region Partition

In order to validate the proposed approach, we have selected some models of mechanical parts to test the efficiency and effect for surface region partition. First, an efficiency curve for surface region partition is given in Fig. 3, where the two coordinate axes respectively describe the face numbers and the decomposition time. From the curve in Fig. 3, we can find the proposed approach takes no more than 1.4 seconds for a CAD model with 600 faces. Because most of mechanical part models only have tens even hundreds of faces, the efficiency of the proposed approach could meet the requirement of practical applications. Second, some experiment results for surface region partition are listed in Fig. 3, where each surface region is assigned a separate color. The results of surface region partition confirm that the proposed approach is feasible and most surface regions obtained have obvious engineering semantics.

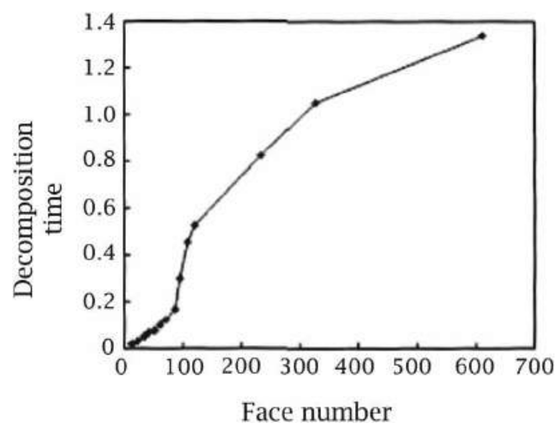


Fig. 3: The efficiency curve for CAD model surface region partition.

4. SURFACE REGION PROPERTIES AND THEIR CODE REPRESENTATION

Generally, the code comparison is faster than the approaches of graph isomorphism checking, and it is also more convenient for B-rep models than the approaches are based on shape statistics and analysis. The reason is that the B-rep models are mainly composed of faces with regular geometry like plane, cylinder, cone, sphere and so on. But topologies of B-rep models in mechanical engineering are determined by their face geometries (plane, cylinder, cone, sphere and so on); if two adjacent faces have the same geometry, they should belong to the same face. Consequently, we adopt a relatively simple approach, which is based on a kind of codes to describe surface region properties.

4.1. Region Header Code

Header code *RegH* describes region convexity and its face type, which is an integer composed in a way as follows:

$$RegH = 32 \times RegC + RegfT,$$

Fig. 4: Surface region partition results for CAD models.

where *RegC* and *RegfT* respectively represent region convexity and its face type. Here, $RegC = 0$, 1 or 2, respectively, if the region is *PR*, $C_V R$, or $C_c R$. Meanwhile, *RegfT* is defined as follows:

$$RegfT = a_0 + 2 \times a_1 + 4 \times a_2 + 8 \times a_3 + 16 \times a_4,$$

where a_i ($i = 0 \sim 4$) is integer 1 or 0. If plane, cylinder, cone, sphere or others respectively exist in a region, a_i is 1; else, a_i is 0.

4.2. Region Relation Code

Here, *RegAdj* represents adjacency edge convexity and adjacency region convexity. It is formulated as follows:

$$RegAdj = 9 \times RegEC + 3 \times RegC_{\min} + RegC_{\max},$$

where *RegEC* is an integer representing adjacency edge convexity, and $RegC_{\min}$ and $RegC_{\max}$ are, respectively, the minimum and maximum value of the codes that describe the two adjacent regions' convexity. Here, *RegEC* are represented in an integer code as follows:

$$RegEC = b_0 + 2 \times b_1 + 4 \times b_2,$$

where b_i ($i = 0 \sim 2$) is integer 1 or 0. If tangent edge, convex edge or concave edge, respectively exists in a region, b_i is 1; else, b_i is 0.

4.3. Face Context Code

Before giving the definition of face context code, the codes for faces and edges are introduced first.

4.3.1. Face properties and their code representations

The face's properties like surface type and face convexity are expressed with a face code *fH*. Here, it is formulated as follows:

$$fH = 10 \times fType + fCon,$$

where *fH*, *fType* and *fCon* are integers, and *fType* or *fCon*, respectively, represents the surface properties mentioned above. Especially, $fType = 0$, 1, 2, 3 or 4, respectively, if the surface is plane, cylinder, cone, sphere, or others, and $fCon = 0$, 1, 2 or 3, respectively, if the surface is planar, convex, concave, or others.

4.3.2. Edge Properties and their Code Representations

Edge properties code *fAdj* represents the edge properties like edge convexity and adjacent surface type. These properties are represented in an integer code as follows:

$$fAdj = 25 \times eCon + 5 \times fType_{\min} + fType_{\max},$$

where $fType_{\min}$ and $fType_{\max}$ are respectively the minimum and maximum value of the codes that express the two adjacent surfaces' geometry types. And *eCon* is an integer representing edge convexity, it is determined by the dihedral angle of the two incident faces; $eCon = 0$, 1 or 2, respectively, if the angle is equal to 180°, less than 180° or others.

4.3.3. Calculation of Face Context Code

First, we use a *FARG* $G_r = (V_r, E_r)$ to represent all faces in a region. Then, G_r is turned into layer *FARG* based on the shortest distance between vertex v_r and other vertices. Meanwhile, the set V_r is divided into k ($k = 1, 2, \ldots, \mathrm{m}$) layers around vertex v_r, v_r is 0 layer L_0 while layer L_k with L_0 has the shortest distance k (see Fig. 5). Last, surface context code *RegfC* is defined as follows:

$$RegfC(v) = fH(v) - C_1 - C_2 - \cdots - C_{\mathrm{m}},$$

$$C_k = 10^4 \times fHL_k + 10^2 \times fAdjI_k + fAdjO_k,$$

$$fHL_k = \left(\sum_{v_i \in L_k} fH(v_i) \right) \Big/ |L_k|,$$

$$fAdjI_k = \left(\sum_{e_i \in L_{k-1} \times L_k} fAdj(e_i) \right) \Big/ |L_{k-1} \times L_k|,$$

$$fAdjO_k = \left(\sum_{e_i \in L_k \times L_k} fAdj(e_i) \right) \Big/ |L_k \times L_k|,$$

where fHL_k, $fAdjI_k$, $fAdjO_k$, $|L_k|$, $|L_{k-1} \times L_k|$ and $|L_k \times L_k|$ are integers, and fHL_k or $|L_k|$ is respectively the average sum of all surface properties code *fH* or its number in layer L_k, and $fAdjI_k$ or $|L_{k-1} \times L_k|$ is respectively the average sum of all edge properties code

$fAdj(e)$ or its number between layer L_{k-1} and L_k, and $fAdjO_k$ or $|L_k \times L_k|$ is respectively the average sum of all edge properties code $fAdj(e)$ or its number in layer L_k.

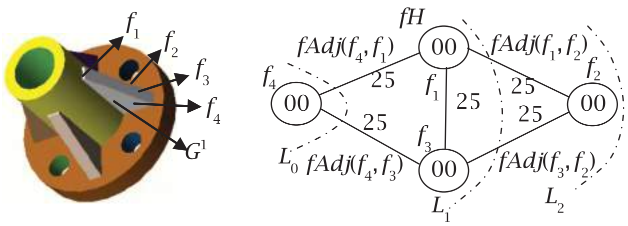


Fig. 5: Layer ARG and its code in the region: (a) CAD model and its region G^1, (b) Face f_4's layer *FARG*.

In Fig. 5 surface region G^1 has four faces f_1, f_2, f_3 and f_4. Face f_4 lies in layer L_0, $fH(v) = 00$. Face f_1 and f_3 lie in Layer L_1, $fH(f_1) = fH(f_3) = 00$, $|L_1| = 2, fHL_1 = 00$. And there are two edges between layer L_0 and L_1, $fAdj(f_4, f_1) = fAdj(f_4, f_3) = 25, |L_0 \times L_1| = 2$, $fAdjI_1 = 25$. Meanwhile, there is only one edge in layer L_1, $fAdj(f_1, f_3) = 25$, $|L_1 \times L_1| = 1$, $fAdjO_1 = 1$. Thus, $C_1 = 002525$. Face f_2 lies in layer L_2, $fH(f_2) = 00$, $fHL_2 = 00$. And there are two edges between layer L_1 and L_2, $fAdj(f_1, f_2) = fAdj(f_3, f_2) = 25$, $|L_1 \times L_2| = 2$, $fAdjI_2 = 25$. No edge is in layer L_2, $fAdjO_2 = 00$. So $C_2 = 002500$. L_3 doesn't exist. Consequently, face f_4's context code is 00002525002500. In the same way, face f_2's, f_1's or f_3's context code is respectively 00002525002500, 00002525, or 00002525.

5. GENERAL RETRIEVAL FOR CAD MODELS

In this paper, general retrieval for CAD models is based on the region code similarity of the two comparing models. In order to compare the multiple codes among different regions conveniently, we need to calculate the number of region with the same code in a model, the boundary edges or faces with the same codes in a region.

5.1. Similarity Assessment for Surface Region Properties

For retrieval model p_r and target model p_t, the statistic results of their region properties are expressed as follows:

$$n_{h,r1} - RegH_{r1}, n_{h,r2} - RegH_{r2}, \ldots, n_{h,ri} - RegH_{ri}, \ldots, n_{h,ra} - RegH_{ra};$$

$$n_{f,ri1} - RegfC_{ri1}, n_{f,ri2} - RegfC_{ri2}, \ldots, n_{f,rik} - RegfC_{rik}, \ldots, n_{f,ric} - RegfC_{ric};$$

$$n_{h,t1} - RegH_{t1}, n_{h,t2} - RegH_{t2}, \ldots, n_{h,tj} - RegH_{tj}, \ldots, n_{h,tb} - RegH_{tb};$$

$$n_{f,tj1} - RegfC_{tj1}, n_{f,tj2} - RegfC_{tj2}, \ldots, n_{f,tjl} - RegfC_{tjl}, \ldots, n_{f,tjd} - RegfC_{tjd}.$$

Where $1 \le i \le a$, $1 \le j \le b$, $1 \le k \le c$, $1 \le l \le d$; a and b are respectively the number of region header code type in p_r and p_t; c and d are respectively the number of face context code type in p_r' region type i and p_t' region type j; and $n_{h,ri}$, $n_{f,rik}$ respectively express $RegH_{ri}$'s number, $RegfC_{rik}$'s number while $n_{h,tj}$, $n_{f,tjl}$ respectively express $RegH_{tj}$'s number, $RegfC_{tjl}$'s number.

When $RegH_{ri} = RegH_{tj}$ and $RegfC_{rik} = RegfC_{tjl}$, the similarity of surface region properties between p_r and p_t can be calculated as the following:

$$SregN(p_r, p_t) = \frac{2 \times \sum_{i=1}^{a} \sum_{j=1}^{b} Sreg_{ri,tj}}{\sum_{i=1}^{a} n_{h,ri} + \sum_{j=1}^{b} n_{h,tj}};$$

$$Sreg_{ri,tj} = \min(n_{h,ri}, n_{h,tj}) \times \frac{2 \times \sum_{k=1}^{c} \sum_{l=1}^{d} \min(n_{c,rik}, n_{c,tjl})}{\sum_{k=1}^{c} n_{c,rik} + \sum_{l=1}^{d} n_{c,tjl}}.$$

Here, if $RegfC_{rik} \ne RegfC_{tjl}$, $\min(n_{c,rik}, n_{c,tjl}) = 0$; $RegH_{ri} \ne RegH_{tj}$, $\min(n_{h,ri}, n_{h,tj}) = 0$.

An example for calculating $SregN(p_r, p_t)$ is given in the following. The two comparison models and their property codes are listed in Table 1.

	RegH	*RegfC*	*RegAdj*
 Retrieval model	4 - 66 3 - 35	4 - 12 2 - 00112600 3 - 11002600 2 - 00112600002600	4 - 23 2 - 40
 Target model	5 - 66 2 - 35 4 - 33	5 - 12 1 - 00112600 2 - 11002600 2 - 00112600002600 8 - 00002525 8 - 00002525002500	6 - 23 9 - 40

Tab. 1: The two comparison models and their property codes.

$$RegH_{r1} = RegH_{t1} = 66, n_{h,r1} = 4, n_{h,t1} = 5, \quad \min(n_{h,r1}, n_{h,t1}) = 4;$$

$$RegfC_{r11} = RegfC_{t11} = 12, n_{c,r11} = 4, n_{c,t11} = 5, \quad \min(n_{c,r11}, n_{c,t11}) = 4;$$

$$Sreg_{r1,t1} = 4 \times \frac{2 \times 4}{4 + 5} = 3.5556.$$

$$RegH_{r2} = RegH_{t2} = 35, n_{h,r2} = 3, n_{h,t2} = 2, \quad \min(n_{h,r2}, n_{h,t2}) = 2;$$

$$RegfC_{r21} = RegfC_{t21} = 00112600, n_{c,r21} = 2,$$

$$n_{c,t21} = 1, \min(n_{c,r21}, n_{c,t21}) = 1;$$
$$RegfC_{r22} = RegfC_{t22} = 11002600, n_{c,r22} = 3,$$
$$n_{c,t22} = 2, \min(n_{c,r22}, n_{c,t22}) = 2;$$
$$RegfC_{r23} = RegfC_{t23} = 00112600002600,$$
$$n_{c,r23} = 2, n_{c,t23} = 2, \min(n_{c,r23}, n_{c,t23}) = 2;$$
$$Sreg_{r2,t2} = 2 \times \frac{2 \times (1+2+2)}{2+1+3+2+2+2} = 1.6667.$$
$$SregN(p_r, p_t) = \frac{2 \times (3.5556 + 1.6667)}{4+5+3+2} = 0.7460.$$

5.2. Similarity Evaluation for Region Adjacency Relation Properties

For retrieval model p_{r} and target model p_{t}, the statistic results of their region adjacency relation properties are expressed as follows:

$$n_{a,r1} - RegAdj_{r1}, n_{a,r2} - RegAdj_{r2}, \ldots, n_{a,rm} - RegAdj_{rm}, \ldots, n_{a,re} - RegAdj_{re};$$
$$n_{a,t1} - RegAdj_{t1}, n_{a,t2} - RegAdj_{t2}, \ldots, n_{a,tn} - RegAdj_{tn}, \ldots, n_{a,tf} - RegAdj_{tf}.$$

Where $1 \le m \le e$, $1 \le n \le f$; e and f are respectively the number of face adjacency relation properties type in p_{r} and p_{t}; $n_{a,rm}$ and $n_{a,tn}$ respectively represent $RegAdj_{rm}$'s number and $RegAdj_{tn}$'s number.

If $RegAdj_{rm} = RegAdj_{tn}$, the similarity of region adjacency relation properties between p_{r} and p_{t} can be calculated as the following:

$$SregAdj(p_r, p_t) = \frac{2 \times \sum_{m=1}^{e} \sum_{n=1}^{f} \min(n_{a,rm}, n_{a,tn})}{\sum_{m=1}^{e} n_{a,rm} + \sum_{n=1}^{f} n_{a,tn}}.$$

When $RegAdj_{rm} \ne RegAdj_{tn}$, $\min(n_{a,rm}, n_{a,tn}) = 0$.

For the two CAD models in Table 1, the similarity $SregAdj(p_{\mathrm{r}}, p_{\mathrm{t}})$ is given in the following.

$$RegAdj_{r1} = RegAdj_{t1} = 23, n_{a,r1} = 4, n_{a,t1} = 6, \min(n_{a,r1}, n_{a,t1}) = 4;$$
$$RegAdj_{r2} = RegAdj_{t2} = 40, n_{a,r2} = 2, n_{a,t2} = 9, \min(n_{a,r2}, n_{a,t2}) = 2;$$
$$SregAdj(p_r, p_t) = \frac{2 \times (4+2)}{4+6+2+9} = 0.5714.$$

5.3. Similarity Assessment for CAD Models

Let $SregN(p_{\mathrm{r}}, p_{\mathrm{t}})$ and $SregAdj(p_{\mathrm{r}}, p_{\mathrm{t}})$ be the similarity of surface region properties and adjacency relation properties between retrieval model p_{r} and target model p_{t} respectively. The similarity of p_{r} and p_{t} is calculated as in the following:

$$Sreg(p_r, p_t) = \omega_1 \times SregN(p_r, p_t) + \omega_2 \times SregAdj(p_r, p_t);$$

where $\omega_1 + \omega_2 = 1, 0 \le \omega_1, \omega_2 \le 1$.

If the two comparing models with similar topology are more important, ω_2 should be set larger. On the contrary, if geometry similarity is more important, ω_1 should be set larger. If $\omega_1 = \omega_2 = 0.5$, the similarity of retrieval model and target model in Table 1 is

$$Sreg(p_r, p_t) = 0.5 \times 0.7460 + 0.5 \times 0.5714 = 0.6587.$$

6. PARTIAL RETRIEVAL FOR CAD MODELS

Here, partial retrieval of CAD models is based on matching surface region codes. The two surface regions are matched if they have the same or similar region codes. Meanwhile, partial retrievals are realized in two ways, rough retrieval and exact retrieval, which are provided for meeting various retrieval requirements. Rough retrieval can find subparts that have a certain similarity to the retrieval subpart. When users are not sure what are their desired reusable subparts and only aim at getting inspiration from retrieved subparts, they may prefer to use the rough retrieval. Exact retrieval is based on all the code matching, which usually returns fewer subparts. It supports users to discover those subparts similar to the retrieval in all aspects.

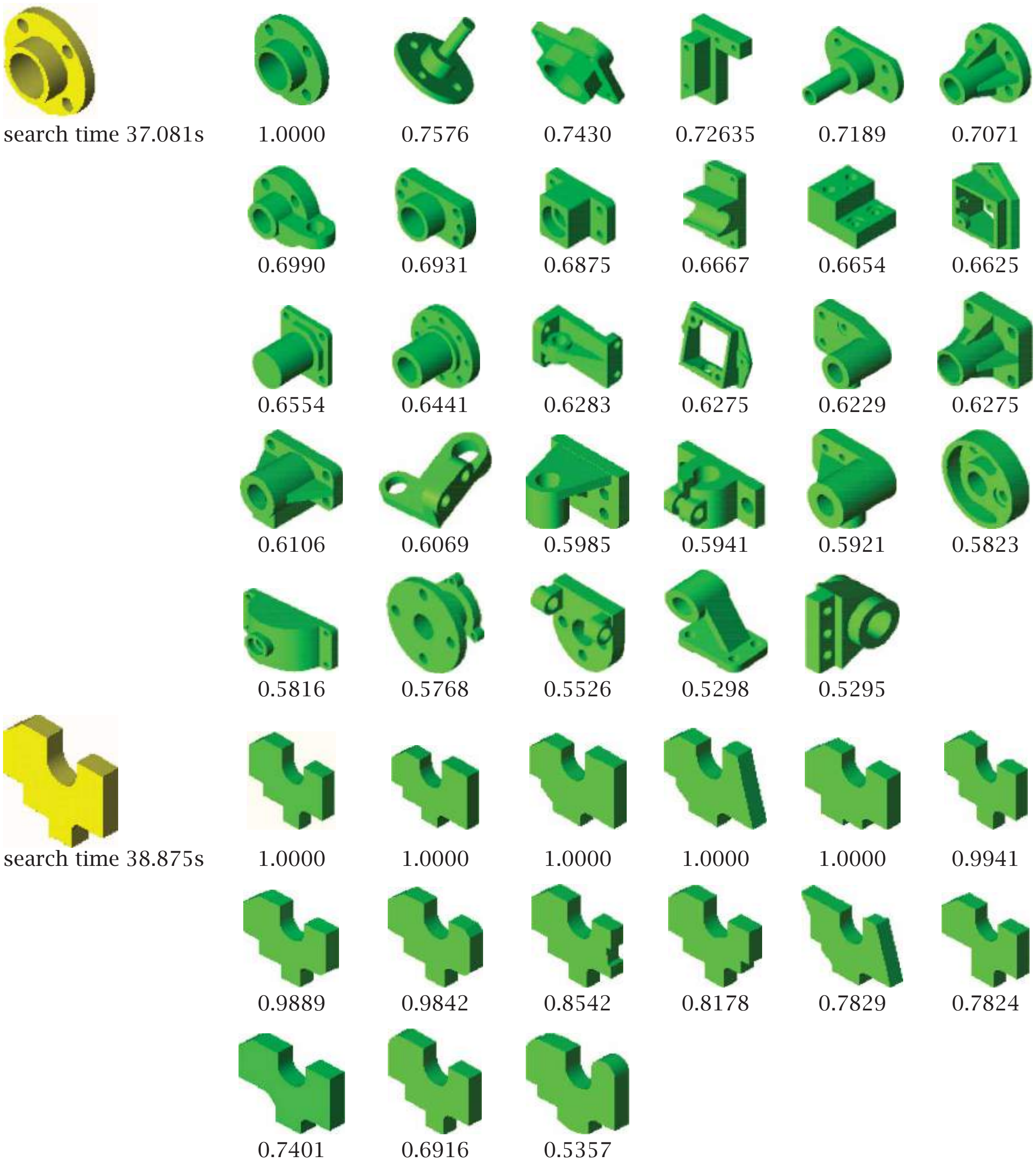
6.1. Region Properties Matching

For retrieval model's region r_r and target model's region r_t, Let $RegH_r$ and $RegH_t$ be their header codes, $RegfC_{ri}$ and $RegfCN_{tj}$ are respectively their face context codes. Their statistic results are expressed as follows:

$$n_{c,r1} - RegfC_{r1}, \ldots, n_{c,r2} - RegfC_{r2}, \ldots, n_{c,ri} - RegfC_{ri}, \ldots, n_{c,rc} - RegfC_{rc};$$
$$n_{c,t1} - RegfC_{t1}, \ldots, n_{c,t2} - RegfC_{t2}, \ldots, n_{c,tj} - RegfC_{tj}, \ldots, n_{c,td} - RegfC_{td}.$$

Where $1 \le i \le c$, $1 \le j \le d$; i and j respectively express the number of face context code type in region r_r and r_t; $n_{c,ri}$ and $n_{c,tj}$ respectively represent $RegfC_{ri}$'s number and $RegfC_{tj}$'s number.

When $RegH_r = RegH_t$, and for each $RegfC_{ri}$ in region r_r, if there is an equal $RegfC_{tj}$ in region r_t, the properties of region r_r and r_t are thought to be matched in the rough mode. In addition to above conditions, if $n_{c,ri} = n_{c,tj}$, their properties are thought to be matched in the exact mode.



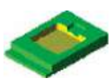
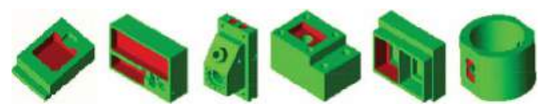
Retrieval model	Retrieval results					
search time 37.081s	1.0000	0.7576	0.7430	0.72635	0.7189	0.7071
	0.6990	0.6931	0.6875	0.6667	0.6654	0.6625
	0.6554	0.6441	0.6283	0.6275	0.6229	0.6275
	0.6106	0.6069	0.5985	0.5941	0.5921	0.5823
	0.5816	0.5768	0.5526	0.5298	0.5295	
search time 38.875s	1.0000	1.0000	1.0000	1.0000	1.0000	0.9941
	0.9889	0.9842	0.8542	0.8178	0.7829	0.7824
	0.7401	0.6916	0.5357			

Tab. 2: Results of some general-model retrievals in the model library.

6.2. Region Adjacency Relation Properties Matching

For retrieval model's region r_r and target model's region r_t, Let $RegAdj_{rm}$ and $RegAdj_{tn}$ be their adjacency relation codes respectively. Their statistic results are expressed as follows:

$$n_{a,r1} - RegAdj_{r1}, n_{a,r2} - RegAdj_{r2}, \ldots, n_{a,rm} - RegAdj_{rm}, \ldots, n_{a,re} - RegAdj_{re};$$

Retrieval models	search time (s)	Retrieval results
	Rough retrieval (36.14)	
	Exact retrieval (36.75)	
	Rough retrieval (37.484)	
	Exact retrieval (33.375)	
	Rough retrieval (38.000)	
	Exact retrieval (37.766)	

Tab. 3: Retrieval results for three typical subparts.

$$n_{a,t1} - RegAdj_{t1}, n_{a,t2} - RegAdj_{t2}, \ldots, n_{a,tn} - RegAdj_{tn}, \ldots, n_{a,tf} - RegAdj_{tf}.$$

Where $1 \le m \le e$, $1 \le n \le f$; m and n are respectively the number of adjacency relation type in region r_r and r_t; $n_{a,rm}$ and $n_{a,tn}$ respectively represent $RegAdj_{rm}$'s number and $RegAdj_{tn}$'s number.

For each $RegAdj_{rm}$ in region r_r, if there is an equal $RegAdj_{tn}$ in region r_t, the adjacency relation properties of region r_r and r_t are thought to be matched in the rough mode. In addition to above conditions, if $n_{a,rm} = n_{a,tn}$, their adjacency relation properties are thought to be matched in the exact mode.

6.3. Region Matching

For retrieval model's region r_r and target model's region r_t, when their region properties and adjacency relation properties are respectively matched in the rough mode, region r_r and r_t are thought to be matched in the rough mode. Meanwhile, if their region properties and adjacency relation properties are respectively matched in the exact mode, they are thought to be matched in the exact mode.

7. EXPERIMENTS RESULTS

In this paper, some experiments for general and partial of CAD models have been conducted to validate the proposed approach. These tests were performed on a computer with an Intel 3.0 GHz CPU and a 1.0 GB RAM. The library is downloaded from multiple online CAD model libraries, which collects 450 feature-rich CAD models with ACIS format. The reason for no using an existing test library of engineering parts is that almost all existing libraries just have mesh models instead of B-rep models, while B-rep models are the dominant models in engineering applications.

In order to test the retrieval effect and efficiency, we try to search for two typical general models and three typical subparts in the library using the program and some retrieval results are found. Since surface region partitions and region code generations are carried out in the offline phase, the search time listed in the table does not include the time for surface region partition and generation region property codes. In addition, they are the average time of 50 runs of the program for model retrieval.

7.1. General Retrieval

Here, the two typical models with 10 and 18 faces are selected for general retrieval. Because the topologies of B-rep CAD models are uniquely determined by their face geometries (plane, cylinder, cone, sphere and so on), the geometry and topology properties are thought to be the same importance. Consequently, we set $\omega_1 = \omega_2 = 0.5$ in the experiment. In Table 2, some retrieval results and search time are listed. For simple, the similarity threshold is set 0.5 here. The experiment results show that the proposed approach supports the general retrieval and the search time is acceptable.

7.2. Partial Retrieval

For partial retrieval, three typical subparts with rich features are selected for evaluating the retrieval effect and efficiency of the program developed based on the proposed approach. In Table 3, the three subparts, respectively having 6, 6 and 9 faces, selected from query models are presented in orange color, while their similar subparts found in the library are shown in red color. Meanwhile, their search time is also listed in the table. Experimental results show that the proposed approach well supports the partial retrieval of CAD models and it is promising to meet the requirement of engineering applications.

8. CONCLUSIONS

Because *FARG* is convenient to be created directly from CAD model's boundary representation and the existing subgraph matching algorithms can be used for the model comparison, some researches selected *FARG* to describe CAD models. But the size of *FARG* in a CAD model may be large and subgraph matching may not be effective and efficient enough because of its NP problem.

In this paper, a general and partial retrieval approach for CAD models is developed by handling models' *FARG*s, which includes surface region partition, Surface region properties and their code representation, and shape comparison based on region shape codes. In order to gain salient geometric features in a model, we use surface region partition to separate regions with different convexity. Although the size of surface regions are smaller than the number of faces, the compassion efficiency of surface regions still needs to be guaranteed with certain measures for practical applications. Here, face context codes combined with the codes reflecting region's convexity and relations are further generated from region's *FARG* to serve as the substitutes of the original *FARG* shape description, and consequently the matching efficiency is achieved by replacing the subgraph isomorphism checking process with the comparison of the codes. Experiments show that the face context codes have good shape differentiating capability. All in all, the proposed approach is an efficient and effective approach for general and partial retrieval of CAD models.

ACKNOWLEDGEMENT

This work was supported by the Scientific Research Project of Hubei Educational Department (B2013229).

REFERENCES

[1] Gunn, T. G.: The mechanization of design and manufacturing, Scientific American 247(3), 1982, 114–130.

[2] Osada, R.; Funkhouser, T.; Chazelle, B.; Dobkin, D.: Shape distributions, ACM Transactions on Graphics, 21 (4), 2002, 807–832.

[3] Kazhdan, M.; Funkhouser, T.; Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors, In: Proceedings of the Eurographics Symposium on Geometry Processing, 2003.

[4] Novotni, M.; Klein, R.: Shape retrieval using 3D Zernike descriptors, Computer-Aided Design, 36 (11), 2004, 1047–1062.

[5] Daras, P.; Axenopoulos, A.: A 3D shape retrieval framework supporting multimodal queries, International Journal of Computer Vision, 89, 2010, 229–247.

[6] Gao, Y.; Dai, Q.; Zhang N.: 3D model comparison using spatial structure circular descriptor, Pattern Recognition, 43(3), 2010, 1142–1151.

[7] Sundar, H.; Silver, D.; Gagvani, N.; Dickenson, S.: Skeleton based shape matching and retrieval, In: Proc. shape modeling international, 2004, 130–139.

[8] Bespalov, D.; Regli, W. C.; Shokoufandeh, A.: Reeb graph based shape retrieval for CAD, In Proceedings of DETC03 2003 ASME Design Engineering Technical Conferences, 2003.

[9] Bespalov, D.; Shokoufandeh, A.; Regli, W. C.; Sun, W.: Scale-space representation of 3D models and topological matching, Proceedings of the eighth ACM symposium on Solid modeling and applications, 208–215, New York, NY, USA, 2003.

[10] Tao, S. Q.; Huang, Z. D.; Zuo, B. Q.; et al.: Partial retrieval of CAD models based on the gradient flows in Lie group, Pattern Recognition, 45 (4), 2012, 1721–1738.

[11] El-Mehalawi, M.; Allen, M. R.: A database system of mechanical components based on geometric and topological similarity, Part I: representation, Computer-Aided Design, 35 (1), 2003, 95–105.

[12] Li, M.; Zhang, Y. F.; Fuh, J. Y. H.; et al.: Toward effective mechanical design reuse: CAD model retrieval based on general and partial shapes, ASME Journal of Mechanical Design, 131 (12), 2009, 121501.1–121501.8.

[13] Bai, J.; Gao, S.; Tang, W.; et al.: Design reuse oriented partial retrieval of CAD models, Computer-Aided Design, 42 (12), 2010, 1069–1084.

[14] Biasotti, S.; Marini, S.; Spagnuolo, M.; et al.: Subpart correspondence by structural descriptors of 3D shapes, Computer-Aided Design, 38(9), 2006, 1002–1019.

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

[16] Saber, E.; Xu, Y.; Tekalp, A.M.: Partial shape recognition by sub-matrix matching for partial matching guided image labeling, Pattern Recognition, 38 (10), 2005, 1560–1573.

[17] Bespalov, D.; Ip, C. Y.; Regli, W. C.; Shaffer, J.: Benchmarking CAD search techniques, In Proceedings of the 2005 ACM symposium on Solid and physical modeling (SPM '05), New York, NY, USA, 2005, 275–286.

[18] Sonthi, R.; Kun, J. G.; Gad, H. R.: Shape feature determination using the curvature region representation, Proceedings of the 4th Symposium on Solid Modeling and Applications Atlanta, 1997, 285–296.

[19] Fu, M. W.; Ong, S. K.; Lu, W. F.; et al.: An approach to identify design and manufacturing features from a data exchanged part model, Computer-Aided Design, 35(11), 2003, 979–993.