Eurographics Italian Chapter Conference (2007)
Raffaele De Amicis and Giuseppe Conti (Editors)

Form Features in Non-manifold Shapes: A First Classification and Analysis

C. Crovetto[1,2,3] L. De Floriani[1] and F. Giannini[2]

[1] Department of Computer Sciences,
University of Genova, via Dodecaneso, 35 - 16146 Italy

[2] Institute of Applied Mathematics and Informatic Technologies - CNR Genova
Via De Marini 6, 16149 Genova, Italy

[3] Soils, Solids, Structures Laboratory, University of Grenoble
Domaine Universitaire BP 53 - 38041 Grenoble Cedex 9 France

Abstract

During the industrial design process, a product model undergoes several analyses. One of the most common ones is the finite element analysis. This kind of analysis needs a simplified model, which can include idealised parts, and thus it is usually non-manifold and non-regular. During the idealisation process, the semantic information attached to the CAD model, such as features or surface types, or information used for model simplification, e.g. assumptions on the behavior type, is usually lost, thus making more difficult re-using, or, at least taking advantage, of performed simulations and models. This would be made easier if a meaningful interpretation of the object sub-parts is available. To this aim, in this paper, we provide a taxonomy of form features in non-manifold shapes and we describe an approach for their identification based on a decomposition of a non-manifold shapes into uniformly dimensional components proposed in [DHH06]. The process presented is the first step towards the identification of form features, since it analyzes those features containing non-manifold singularities.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Applications]: Feature models, Product design and analysis, Non-manifold representations

1. Introduction

The complete and detailed specification of a product shape is one of the major outcomes of the development process of new products. It is the result of many behavior simulation and reasoning processes typical of the different phases of the product life cycle. These processes are needed in order to guarantee that the product fulfils the functional requirements and constraints defined on the basis of client specifications [Lee99].

Every kind of analysis is associated with a specific engineering field that requires specialist knowledge and specific attributes attached to the various parts forming an object. For this reason, it is necessary to have a different model of the same object for every view.

The types of attributes attached with the various models are not only geometric, but also qualitative. Therefore, all the representations commonly used in CAD systems based on primitive entities, like vertices, edges and faces, as in the case of boundary representations [Man87], or composing volumes, such as cylinders, cones and spheres for the Constructive Solid Geometry [RT78], are inadequate. They provide a description of the object just in terms of low-level geometric entities not meaningful from the application point of view.

A possible solution to these limitations is to use a feature-based representation which provides a structuring of the underlying geometric model [SM95]. Features, in fact, are used to associate an engineering meaning with a part, or a set of parts of an object.

Moreover, in some simulation analyses, such as stress analysis, we have to deal with additional problems related to the execution time. The time needed to solve the system of equations that compute a given analysis is too high when the model is complex. Therefore, to perform the computation in a reasonable time, a representation that contains only data really affecting the behavior under analysis is needed. For this reason, the model used during the first design phase cannot be used and a new simpler model has to be generated. Usually, the required structure for the various analyses is a finite set of interconnected meshes. The activity of creating these meshes is called Finite-Element Model (FEM) preparation. This activity comprises the specification of the boundary conditions, that are the various forces acting on the object and external loads attached to the object.

To have a simpler model, some shape details which do not affect simulation results are eliminated and some parts of the object are idealized. For example, if a 3D part of the object presents a behavior comparable to that of a beam, that 3D object sub-region can be replaced by a 1D element. Similarly, if its behavior is analogous to that of a lamina it can be replaced by a 2D element [Gro06]. After the idealisation process, the structure can be composed by parts of different dimensions, therefore it can be non-manifold.

Several methods have been proposed for such shape adaptation process of CAD models for the Finite Element Analysis (FEA). Among them, in [BBT*99], a morphological feature recognition approach has been adopted to identify features that might be suppressed, or idealised, and to maintain consistency between the original CAD model and the idealizations used for the various views. Similar approaches can be found in the works of Hamri et al. [HLG04] and of Lee [Lee05]. The first propagates CAD information in the tessellated representation and combines a feature-based simplification with a shape adaptation guided by mechanical criteria directly on the mesh representation. Whereas, Lee exploits the feature information provided by a design-by-feature approach to define idealised parts. All these works highlight the strong link between the presence of features and the idealisations to be performed. Anyhow, generally no relationship between the original CAD model and the idealized version is maintained. It could be useful to evaluate different simplifications of the same shape, or to better understand if after changes in the CAD data a new simulation has to be performed.

In the literature, many studies on feature recognition on manifold objects have been devised [SAKJ01], but very few are dealing with non-manifold objects such the one presented in [FG92]. In this paper, we propose a feature taxonomy for non-manifold shapes and an approach for classifying the various components of a non-manifold shape, which we consider obtained by simplification and idealisation process. At first, the object representation, described by a simplicial 2-complex embedded in 3D Euclidean space, is decomposed into manifold parts attached at singular vertices and/or edges, then each part is classified according to the defined taxonomy by analysing its relationships with the adjacent parts.

The work here described is a first step for assessing the similarity of FEA meshes for the simulation point of view. The remainder of this paper is organized as follows. In Section 2, we briefly describe the decomposition of an object into manifold parts on which our classification is based. In Section 3, we describe the taxonomy of the selected features. In Section 4, we present our classification method. In Section 5, we draw some concluding remarks.

2. A decomposition into manifold parts

In this Section, we review some basic notions about two-diemnsional Euclidean simplicial complexes, and we briefly describe a topological decomposition of a non-manifold shape introduced in [DHH06], that we use as the basis for our form feature classification algorithm.

We consider a set V_σ of $k+1$ linearly independent points in the n-dimensional Euclidean space R_n, with $k \leq n$. The subset σ of R_n formed by the points x that can be expressed as the convex combination of the points of V_σ is called a *k-simplex* generated by V_σ. Note that a 0-simplex is a *vertex*, a 1-simplex an *edge*, a 2-simplex a *triangle*. Any s-simplex τ generated by a subset of V_σ of cardinality $s+1 \leq k$ is called an *s-face* of σ. A finite collection Σ of simplexes is an *Euclidean simplicial complex* if and only if for each simplex σ in Σ, all faces of σ belong to Σ, and, each pair of intersecting simplexes intersect at a common face. If d is the maximum of the orders of the simplexes of Σ, then Σ is called a *d-simplicial complex*. Note that a simplicial 2-complex is formed by vertices, edges and triangles, while a simplicial 1-complex just by vertices and edges.

We consider a non-manifold shape discretized as a simplicial 2-complex embedded in 3D Euclidean space. The decomposition is based on three classes of components namely:

- *wire-webs*, which are maximal connected subcomplexes formed only by wire-edges (i.e. edges which are not on the boundary of any triangle);
- *sheets*, which are maximal 1-connected sub-complexes with boundary formed by two-sided triangles;
- *shells*, which are maximal 1-connected sub-complexes without boundary which include empty volumes.

Note that a 1-connected complex is a simplicial complex in which every pair of triangles or edges can be connected through a path formed only by triangles and edges.

In [DHH06], an algorithm is described for extracting the above components from a simplicial 2-complex. Note that each component in the decomposition may contain non-manifold singularities. Wire-webs may contain only non-manifold vertices, while the other two may contain also non-manifold edges.

3. Our Feature Classification

From the decomposition described above we just obtain information on the shared non manifold entities, whereas for additional considerations, more meaningful information have to be available in order to describe the object composition.

Let us consider the example in Figure 1. The object is decomposed into 6 manifold components: component A, that can be considered the base on which other components are added; components B, D and E can be seen as protrusions of A, component C is an handle attached to A, and finally F connects D and E. This is the most natural interpretation we can give to the components and in fact, it is the interpretation we want to find with our classification algorithm.

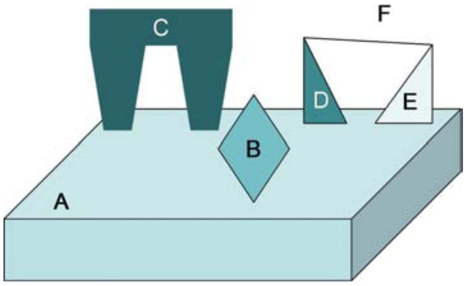

Figure 1: *Components individuation*

We should notice that this is not the unique possible interpretation. For example, in Figure 1, components D, E and F can be considered together as a handle. The various interpretations can be obtained by using a local or a global classification approach. The local one consists in analysing only the relationships of a component with its directly adjacent ones, whereas the global one looks at their concatenation.

Here, we focus on the first approach, which is conceived as the basis for providing the global classification. The component classification in terms of feature we have introduced is based on the form feature taxonomy of STEP (Standard for Exchange of Product Data, ISO 10303), in particular on the subset relative to volume features. The features which we consider in STEP are subdivided in two main categories: added volumes and subtracted volumes. The feature correspondents to added volumes are also divided in *protrusions*, *connectors*, *handles* and *standalones*, while those corresponding to subtracted ones are distinguished in *cavities* and *through holes*.

Since we have to deal with idealized models, that is non-regular and non-manifold models, we will deal not only with volumes but also with 2D and 1D components, i.e. in general of simplicial complexes embedded in R^3.

Since the models we work with on are idealized, it is even more complex to find the types of features to search for, and how they are defined. The problem is that the process of idealization can alter the type of connection between different components of an object. As an example, the object in Figure 2.a is formed by two components A and B connected by the component C, while the object in Figure 2.b is composed by one components A with a protrusion C. These two objects can have the same idealization, shown in Figure 2.c, in which we loose the type of connection between the parts of the object, since we cannot know if the components A and B were touching before idealization or not.

Since we do not have any limitation on the possible objects, we cannot foresee the type and the number of relations between the components. Thus, we have chosen to have relatively few feature types, but with the possibility of assigning more than one type to a single component. In this way, no possible interpretation for a component is lost.

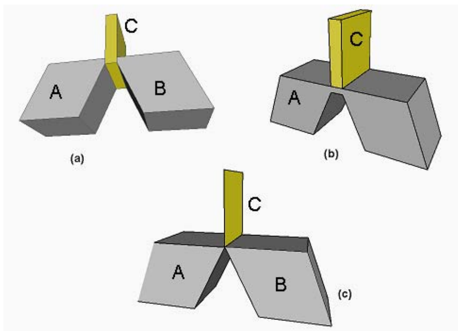

Figure 2: *Two different objects (a) and (b), having the same idealization model (c).*

In the following feature definitions, we consider a "part" as a connected and compact subset of the object that has a meaningful semantics in the application context. In Figure 3, we have an example of an object containing all the features that we consider. Therefore, our classification is as follows:

- **Connector**: part of the object that, if removed, breaks the object in two or more connected parts.
- **Handle**: part of the object that intersects only one other part in two or more ends and is outside of this other part
- **Protrusion**: part of object that intersects only one other part in one end and is outside this other part.
- **Through hole**: part of object that intersects only one another part in two ore more ends and is inside this other part.
- **Cavity**: part of object that intersects only one other part in one end and is inside this other part.
- **Standalone**: part of the object that is not a feature of any other part and can be considered as autonomous.

4. Feature interpretation

To classify the parts, it is necessary to define the properties of each single component and the nature of its relations with

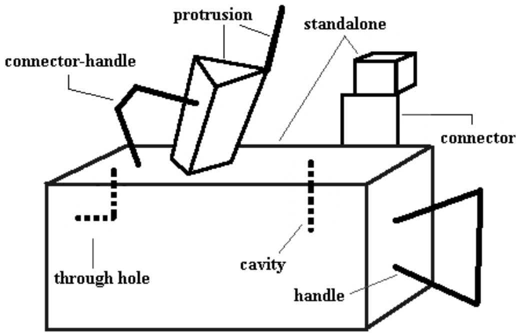

Figure 3: *Examples of the considered features: protrusion, connector, handle, standalone, through hole, cavity.*

its adjacent parts, like the component dimension and the connection type. We call components the sub complexes C_i determined by the decomposition.

4.1. Component dimension

We distinguish among:

- one-dimensional component
- open two-dimensional component
- closed two-dimensional component

A *one-dimensional component* is composed only of vertices and edges.

A *two-dimensional open component* is a two-dimensional simplicial complex with boundary.

A *two-dimensional closed component* is a two-dimensional simplicial complex without boundary.

Based on the component dimension, we can create a first hierarchy between the components, that will help us in their classification. If a two-dimensional component is connected with a one-dimensional component we assume that the lower-dimensional component is a feature of the higher-dimensional component one. Similarly, if we have two two-dimensional components, one open and one closed, the open one will be considered a feature of the closed one.

4.2. Cut between two components

Entities shared by two components C_1 and C_2 are organized in sets called *cut*, to define the type of the connection between C_1 and C_2. The list of *cuts* that connect C_1 to C_2 could be different from the list of *cuts* that connect C_2 to C_1, so we need to consider these two different lists. Let us consider, for example, the objects in the Figure 4 and 5. Both models are composed of two-manifold components. In the first one, the two components are joined by three *cuts*: one consists of the vertex v and the others two of the chains of edges $[e_1, e_2]$ and $[e_3, e_4]$. In this case, we can say that A is connected to B through the same cuts with which B is connected to A. In Figure 5.a, the situation is different. In this case, we can see that component A has the shape of a handle attached to component B by two chains of edges: $[e_1, e_2]$ and $[e_3, e_4]$. On the contrary, component B is not a handle and the chain of edges through which it is joined to A is only one: $[e_1, e_2, e_3, e_4]$. In order to capture the difference between the connection types of the two components, we define the *extended star of a cut*. We recall that, in a simplicial complex Σ, the *star* of a simplex σ is defined as $\{\xi \in \Sigma \mid \sigma \text{ is a simplex on the boundary of } \xi\}$.

Let C_1 be one component and let w be a connected chain of edges. Then, the extended star of w is defined as follows: $\text{st}(C_1, w) = \{\xi \in C_1 \mid \xi \cup st(\sigma), \sigma \in w\}$.

In Figure 5.b, we can observe in darker colours the two extended stars of the chains connecting the two components. While the extended star inside B constitutes a set of 1-connected entities, the star extended inside A is composed by two sets of 1-connected entities. This implies that component A has two different joints with B, and so it is a handle, and component B has only one joint with A, so it is not a handle.

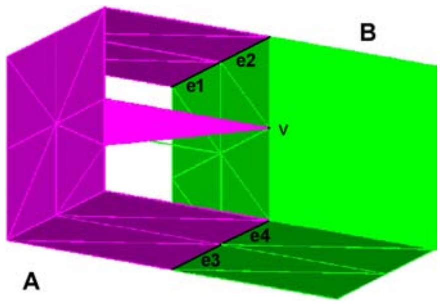

Figure 4: *Components A and B are connected by three cuts: one is vertex v, the other two consist of the chains of edges $[e_1, e_2]$ and $[e_3, e_4]$*

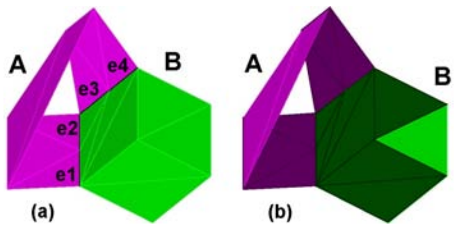

Figure 5: *(a) First special case of a chain of edges that has to be divided (b) We can see in darker colours the two stars of the chain of edges shared by the two components.*

The special cases of *cuts* shown in Figure 5 and 6 are useful to understand the definition of *cut* we give. A *cut* cannot be only defined as the maximal chain of entity shared by two components, because in some cases, a single chain would not have been broken, like in Figure 5. Note that the chain of edges inside the component A of the Figure 5 has been cut in correspondence of the only non-manifold vertex of A. Not always the separation happens in correspondence to non-manifold vertices. Let us consider the example in Figure 6: the entities shared by the two components E and F are the edges e_1, e_2, e_3. We consider the *cuts* that connect F to E. If the division were made in correspondence of non-manifold vertices of F, we would obtain 3 chains composed of a single edge: [e_1], [e_2] and [e_3]. This would not be correct. In fact, member F is clearly one handle that is closed on E with its two partially coincident extremities: the first is composed of edges [e_1, e_2, e_3] and the second one of single [e_2].

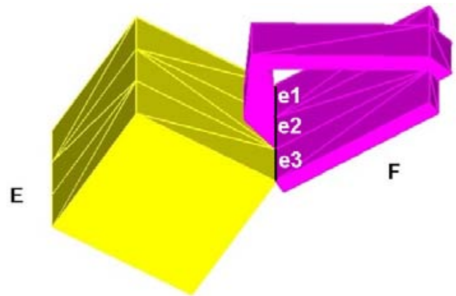

Figure 6: *Second case in which a chain must be divided.*

In these examples, it has been shown the need to maintain two distinguished lists of *cuts* for every pair of components sharing entities. This provides therefore more complete and correct information that allows a better classification of the components. Let us consider the simplicial sub-complex as follows: we indicate with st(σ) the union of the simplexes incidents in σ (star of σ). Let C_1 and C_2 be the two components that have in common one or more vertices or edges and let w be a connected chain of edges [$e_1, e_2, \ldots, e_k$] such that $e_i \in C_1 \cap C_2$ for $i \geq 1$, $i \leq k$.

The cut from C_1 towards C_2 is defined as follows:

- A vertex $v \in C_1 \cap C_2$ belongs to the cut if and only if no edge exists in $C_1 \cap C_2$ having v as an extreme vertex.
- A connected chain w of edges $w = [e_1, e_2, \ldots, e_k]$ belongs to the cut if and only if:
  1. for every edge e_i in w, $e_i \in C_1 \cap C_2$
  2. every pair $e_i, e_j \in w$, $i \neq j$, is 1-connected by means of a path belonging to st (C_1, w) that does not contain any edge in w except e_i and e_j that are the ends of the path.
  3. w is maximal, that is, w does not belong to any longer chain w' that comprises edges [$e_1, \ldots, e_n$] and is connected through the same simplexes of the paths [$h_1, \ldots, h_{n-1}$].

4.3. Classification Algorithm

The feature classification is based on the analysis of the attributes of the components and of the relations with adjacent components. The components adjacent to a given component C are those sharing at least a non-manifold vertex or edge with C. Note that we characterize every component C in relation to its adjacent components of larger or equal dimension. For example, if we want to know the component which can be identified as protrusions, we need to search for components connected to only one other component of larger dimension. Note that such components may be connected or not with a component of lower dimension. Given a component C, we denote with:

- $i_1(C)$ the number of components adjacent to C which are of higher dimension than C and such that there is only one cut from each of such components towards C;
- $i_2(C)$ the number of components adjacent to C which are of higher dimension than C and such that there exists two or more cuts from each of such components towards C;
- $o_1(C)$ the number of components adjacent to C which have the same dimension as C and such that there is only one cut from each of such components towards C;
- $o_2(C)$ the number of components adjacent to C which have the same dimension as C and such that there exists two or more cuts from each of such components towards C.

In the following, we provide the rules for the positive features (handle and protrusion) but the same rule can be applied to the corresponding negative ones (holes and through holes). The distinction is only given by checking if the feature component is internal or external to the component to which it refers to. Moreover, we have added a further specification to the connector feature type to better characterise it. Since the models that we are going to classify are composed of components of different dimension, we choose to use this information as well. Therefore, we classify almost as a connector all the components C_k that intersect at least two components C_x e C_y of higher dimension. We call such components *handle-connector*.

Standalone: A component C is called standalone if it is not a feature of any other component and can be considered as autonomous. Thus,

- $i_1(C) = 0$ and $i_2(C) = 0$

Connector: A component C is called a connector if and only if its removal separates the object into two parts. The condition is as follows

- $i_1(C) + i_2(C) + o_1(C) + o_2(C) \geq 2$ and removing C disconnects the object

Handle-connector: A handle-connector is defined as a mono-dimensional component that intersects at least two two-dimensional closed components. If its removal disconnects the object, then it is no more considered handle connector, but connector. Thus,

- $i_1(C) + i_2(C) \geq 2$ and the removal of C does not leave the object disconnected

Handle: A node C is classified as handle if it intersects another component in two or more ends. Therefore,

- $i_2(C) > 0$
- $i_1(C) = 0$ and $i_2(C) = 0$ and $o_2(C) > 0$

Protrusion: A component C is a protrusion if it intersects only in one end only one component of larger dimension or no components of larger dimension but a component of equal dimension in only one end. Thus,

- $i_1(C) = 1$ and $i_2(C) = 0$
- $i_1(C) = 0$ and $i_2(C) = 0$ and $o_1(C) = 1$ and $o_2(C) = 0$

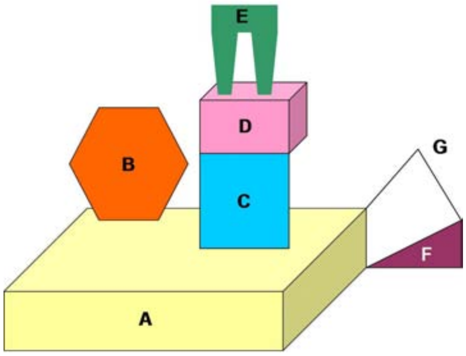

Figure 7: *Example of classification. Component A is classified as standalone. Component B is classified a protrusion, component C is classified as a connector, component D is classified as standalone and a connector, component E is classified as a protrusion and a handle, component F is classified as a protrusion and component G is classified as a connector-handle.*

Moreover, we say that the connection between two components C_i e C_j is *degenerate* if there exists a cut connecting C_i to C_j that is partially or totally coincident with a cut that connects C_i to a third component C_k. Clearly, the connection between C_i to C_k will be degenerate as well. An example is shown in Figure 8, where the components A, B and C are connected by the same vertex v. In this case the connection between the three components A, B and C is degenerate.

5. Concluding Remarks

We have presented an algorithm that produces a classification of form features in a non-manifold and non-regular model. These types of models are used in the analysis phase of a product in its cycle of industrial production.

Their drawback is that the semantics of the various parts and of their reciprocal relationships is lost. Therefore, every time we have to do the analysis on a certain object, we cannot re-use results done in the past on similar objects, because

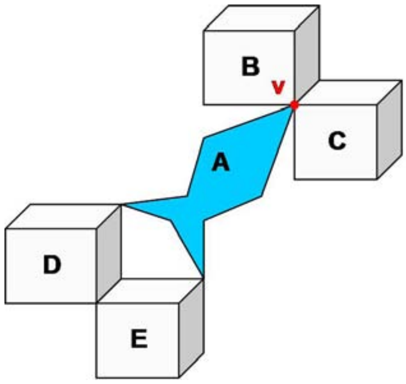

Figure 8: *Examples of degenerate connections: connection between components A, B and C.*

we are not able to know if two objects are similar from the simulation point of view. A features-based model will help in understanding such a similarity.

What we have presented here constitutes a first step in the development of a meaningful classification process. A more global classification is foreseen as future work to allow a better understanding of the overall object organisation.

5.1. Acknowledgements

This work has been supported by AIM@SHAPE European NoE (contract IST 506766) and by project FIRB SHALOM (contract RBIN04HWR8). The authors wish to thank Bianca Falcidieno and Jean-Claude Leon for the many interesting discussions, and Annie Hui for providing the decomposition program and for her help in the development of this work.

References

[BBT*99] BOURAS A., BELAZIZ M., TEHARI A., WAHU G., BRUN J. M.: A strategy to support application's dependent features interpretation. *Journal of Intelligent Manufacturing 10*, 1 (1999), 41–47.

[DHH06] DEFLORIANI L., HUANG M., HUI A.: *Extracting topological characteristics from simplicial non-manifold shapes*. Tech. Rep. CS-TR-4798, Dept. of CS, University of Maryland, College Park, 2006. submitted for publication.

[FG92] FALCIDIENO B., GIANNINI F.: Shape feature decomposition of regularized objects. In *Proccr Visual computing : integrating computer graphics with computer vision* (1992), Kunii T., (Ed.), pp. 481–494.

[Gro06] GROUP F. E. M.: Mathematical coupling of 1d beams to 3d bodies for finite element models, 2006. http://sog1.me.qub.ac.uk/Research/coupling/3d-1d/3d-1d.php.

[HLG04] HAMRI O., LEON J. C., GIANNINI F.: A new approach of interoperability of cad and simulation models. In *TMCE 2004* (2004).

[Lee99] LEE K.: *Principles of CAD/CAM/CAE Systems*. Addison Wesley, 1999.

[Lee05] LEE S.: A cad-cae integration approach using feature-based multi-resolution and multi-abstraction modelling techniques. computer aided design. *Computer Aided Design 37*, 9 (2005), 941–955.

[Man87] MANTYLA M.: *An Introduction to Solid Modeling*. Computer Science press, 1987.

[RT78] REQUICHA A., TIVOLE R.: *Mathematical foundations of constructive solid geometry*. 1978.

[SAKJ01] SHAH. J., ANDERSON D., KIM Y. S., JOSHI S.: A discourse on geometric feature recognition from cad models. *J. Comput. Inf. Sci. Eng. 1*, 1 (2001), 41–51.

[SM95] SHAH J., MANTYLA M.: *Parametric and Feature-Based CAD/CAM : Concepts, Techniques, and Applications*. Wiley-Interscience, 1995.