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

This is an author-deposited version published in: https://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/11388

To cite this version :

Franca GIANNINI, Cédric PETTON, Jean-Philippe PERNOT - Thin part identification for CAD model classification - Engineering Computations - Vol. 31, n°1, p.62-85 - 2015

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

Thin part identification for CAD model classification

Jean-Philippe Pernot
Laboratory LSIS – UMR CNRS 7296, Arts et Métiers ParisTech, Aix-en-Provence, France
Franca Giannini
Istituto di Matematica Applicata e Tecnologie Informatiche, Consiglio Nazionale delle Ricerche, Genova, Italy, and
Cédric Petton
Laboratory LSIS – UMR CNRS 7296, Arts et Métiers ParisTech, Aix-en-Provence, France

Abstract

Purpose – The purpose of this paper is to focus on the characterization and classification of parts with respect to the meshing issue, and notably the meshing of thin parts difficulty handled automatically and which often requires adaptation steps. The objective is to distinguish the so-called thin parts and parts with thin features from the other parts.

Design/methodology/approach – The concepts of thin part and part with thin features are introduced together with the mechanisms and criteria used for their identification in a CAD models database. The criteria are built on top of a set of shape descriptors and notably the distance distribution which is used to characterize the thickness of the object. To speed up the identification process, shape descriptors are computed from tessellated parts.

Findings – A complete modular approach has been designed. It computes shape descriptors over parts stored in a directory and it uses criteria to distinguish three categories: thin parts, parts with thin features and other parts. Being the three categories identified, the user can spend more time on the parts that are considered as more difficulty meshable.

Research limitations/implications – The approach is limited to the three above mentioned categories. However, it has been designed so that the values corresponding to the shape descriptors and associated meshing qualities can easily be inserted within a machining learning tool later on.

Practical implications – The use of the developed tool can be seen as a pre-processing step during the preparation of finite element (FE) simulation models. It is automatic and can be run in batch and in parallel.

Originality/value – The approach is modular, it is simple and easy to implement. Categories are built on top of several shape descriptors and not on a unique signature. It is independent of the CAD modeler. This approach is integrated within a FE simulation model preparation framework and help engineers anticipating difficulties when meshing CAD models.

Keywords CAD models classification, Distance distribution, Shape characteristics and descriptors, Thin parts and parts with thin features

Paper type Research paper

1. Introduction

Nowadays, the evolution toward a full digital simulation of the entire product lifecycle comes with an increase in the amount of digital data thus manipulated. Within such a developmental framework, geometric representations play a key role since they are

The work has been partially supported by the VISIONAIR project funded by the European Commission under grant agreement 262044, and by the project MIUR Fabbrica del Future SuFSeF Sustainable Factory SEmantic Framework.

the privileged intermediate representations shared by most of the actors. Hence, large databases of CAD models are becoming available, thus inaugurating new challenges for their management. This includes the indexing and sorting of those elements so that to retrieve them efficiently at a later time. For example, the complete digital mock-up of a submarine is made up by more than one million of parts representing several terabytes of data.

Unfortunately, neither the actual Product Data Management (PDM) nor Product Lifecycle Management (PLM) systems are good at indexing and sorting CAD models according to their content, i.e. according to the shape and potentially associated semantics in a given context. This is why only some low-level queries can be performed on such databases, and this strongly limits the possibilities to re-use models from past studies. For this reason, there is an increasing need to classify and retrieve parts using parameters which directly exploit shape characteristics (Paquet *et al.*, 2000). Even the shape might be misleading or not enough to discover the desired models. Depending on the purposes, engineers might be interested in parts, which can be very different in their overall shape, but which are similar according to some characteristics or behaviors. It is thus necessary to provide tools enabling context-based shape classification and retrieval. Such a classification can then provide a good means for the re-use of past experiences and knowledge related to specific products or steps of the design process. In this paper, we address the way CAD models can be classified according to the meshing issue.

The product behavior analysis is carried out following three main steps: CAD model definition, CAD model meshing and Finite Elements (FE) simulation. During the two first steps, several adaptations can be manually or semi-automatically handled but they will result in time-consuming modifications:

- The CAD model adaptation step aims at preparing the geometric model for meshing and simulation while performing dedicated geometric transformations. For example, the defeaturing of a CAD model can significantly reduce the meshing and computation times by removing unimportant features (e.g. fillets, holes) which do not significantly affect the simulation results.
- The mesh adaptation step aims at improving the quality of the produced mesh while repairing elements which were badly meshed using operations such as removal, swap, split and so on. The quality of the mesh may also be improved while refining the mesh, i.e. while increasing the number of elements locally.

Both steps require great skills and a deep knowledge in the analysis of where and how the adaptations have to take place. Moreover, they are often performed in loop for several times thus slowing down the preparation process of the simulation model even more.

Having tools to classify parts according to the type of required adaptations could help the engineers to prepare the simulation model better and faster while anticipating the CAD model and mesh adaptation steps. As a matter of fact, having tools to distinguish parts that have to be manually adapted and/or repaired from those that can be automatically meshed would speed up the simulation model preparation steps. Here, the proposed classification method distinguishes three classes: thin parts, parts with thin features, and other parts which do not belong to the first two classes. The use of the thickness of the parts as the main classification criterion is a mean to anticipate the problems occurring when generating tetrahedral meshes from CAD models with

different thicknesses and particularly when a thin feature is linked to a larger volume. As a matter of fact, such configurations may generate a bad quality meshing which is often responsible for inaccuracies during the simulation. This issue is illustrated on Figure 1 where the tetrahedral meshes of two slightly different configurations of a rocker arm are compared to the aspect ratio of the generated elements (Bern and Plassman, 2000). It is admitted that an element with an aspect ratio smaller than 0.5 can be considered as a "bad" element with respect to the accuracy of the final simulation results. The two versions of the rocker arm have been meshed with the same parameters, i.e. same element target size and same chord length error. It is clear that, with about, respectively, 30 and 8 percent of the elements having an aspect ratio smaller than 0.5, the thinnest rocker arm generates skinnier elements (Figure 1(a_1)-(c_1)) than the thickest one (Figure 1(a_2) to (c_2)). Since this process is not fully automated, engineers still have to manually adjust many control parameters. For instance, to improve the quality of the elements resulting from the meshing of the thin rocker arm, they may have to adjust the target size and chord length error initially suggested by the software without considering thin features. Mesh refinements can also be foreseen to adapt the size of the elements to the local configurations.

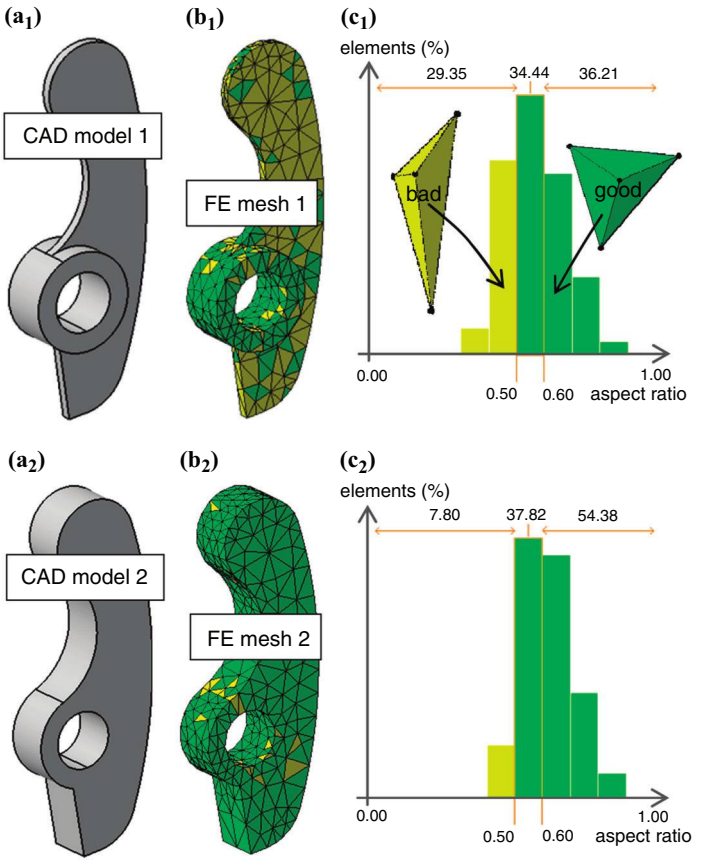


Figure 1. Analysis of the aspect ratios (c_1, c_2) of two FE tetrahedral meshes (b_1, b_2) generated from so-called thin and normal CAD models (a_1, a_2) using the same chord length error and target size

Thus, having tools to classify CAD models according to the type of adaptation and meshing issues could help engineers in better anticipating and preparing the forthcoming treatments. In this paper, we do not address the way in which CAD models can be prepared or the way meshes can be repaired and optimized. Again, the objective is to classify CAD models according to the difficulties the engineers may encounter when meshing them. The classification is built on top of a thickness criterion which distinguishes thin parts, parts with thin features and other parts not belonging to the first two first classes. In this way, designers will spend more time to treat critical models on which thin configurations have been highlighted. Moreover, by using this approach, they may also look for already applied solutions by retrieving parts belonging to the same class.

Furthermore, it is important to say that, even if they may also be used to anticipate and prepare the adaptation steps, the FE semantics (e.g. boundary conditions, material behavior laws) associated to the geometric models have not yet been used to classify the objects. The proposed shape-oriented classification is based on criteria built on top of shape descriptors. Such perspectives are discussed in the conclusions.

In this paper, we extend the approach first presented in Pernot *et al.* (2012) to better take into account the resulting aspect ratios in the identification of the thresholds that drive the classification criteria. The paper is organized as follows. Section 2 introduces the proposed classification approach as well as the state-or-the-art of the existing methods. The considered object classes are presented in Section 3 and the shape descriptors used to characterize those objects are shown in Section 4. The way those shape descriptors are combined to define higher-level object categories is developed in Section 5. Lastly, Section 6 presents the achievements and results we obtain using our prototype software.

2. Related works

In recent times, 3D shape retrieval and clustering have gained big attention due to the fact that large databases of 3D data are becoming available in various fields. Various shape matching methods exist (Tangelder and Veltkamp, 2008; Cardone *et al.*, 2003; Iyer *et al.*, 2005; Li *et al.*, 2010; Brière-Côté *et al.*, 2012; Demirci *et al.*, 2008; Zhu *et al.*, 2012; Ohbuchi *et al.*, 2005) and can be classified in three main categories: feature-based methods, graph-based methods and geometry-based methods (Figure 2). These methods have been implemented and tested and the results are more or less efficient but only when trying to retrieve a shape which is similar to the one of the query from the point of view of form and structure. Even if the problem of shape similarity analysis has been already faced in the CAD/PLM field to solve different problems, the efficient information retrieval is still considered a big challenge in most of the manufacturing companies. In Brière-Côté *et al.* (2012), a survey on the application usage of CAD model comparison is presented and put in correspondence with the solution approaches adopted, namely shape retrieval, similarity assessment and difference identification. Shape similarity and retrieval in CAD/PLM are exploited to reuse product information, in CAD models' management to avoid part duplication, to verify data translation, to manage changes and for product rationalization and standardization. In the first case, the idea is to improve the design process by reusing either the existing parts in a new design with few possible changes or the related associated knowledge and processes (Bai *et al.*, 2010; Bespalov *et al.*, 2005; Chu and Hsu, 2006; Cuillière *et al.*, 2011; Hong *et al.*, 2006; Li *et al.*, 2010; Msaaf *et al.*, 2007; Li *et al.*, 2011; Paquet *et al.*, 2000). To this aim, clustering objects according to

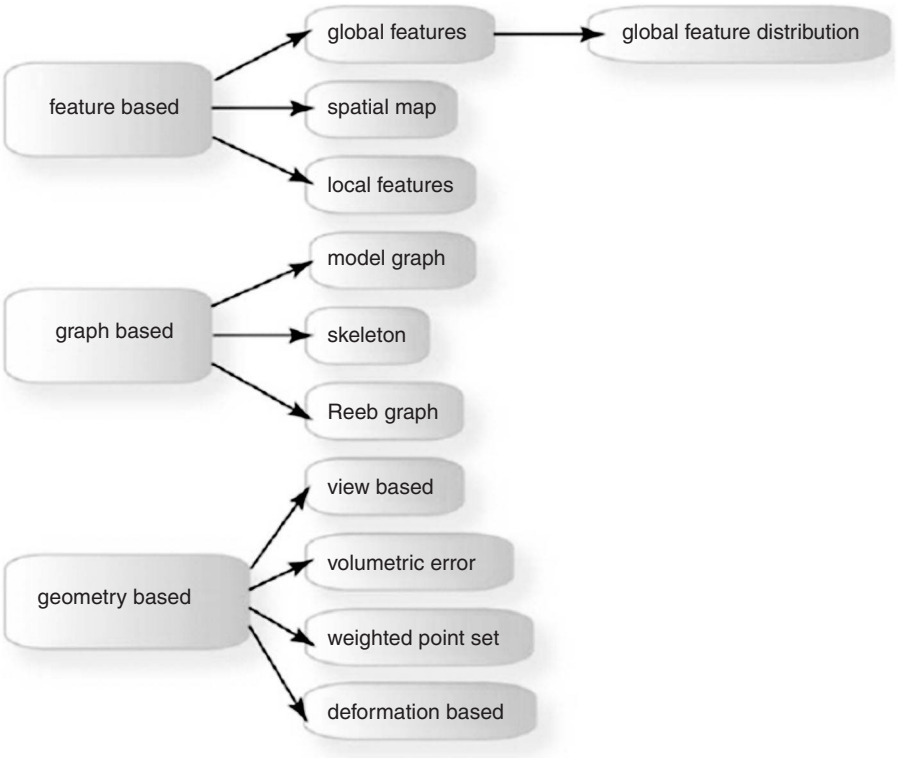


Figure 2.
Taxonomy of shape matching methods

Source: Tangelder and Veltkamp (2008)

specific criteria can allow designers to easily brows repositories and locate similar designs. In this way, designers can take advantage of previous experiences, thus histories of problems encountered in designing and manufacturing of a given group of parts or products can be analyzed for effective decision making and improving new product development (Jayanti *et al.*, 2009). Group technology represents a well-known and successfully applied application-oriented clustering method based on design and manufacturing attributes.

The work presented here deals with the problem of clustering 3D CAD models according to categories presenting common characteristics from the FE mesh generation point of view. If compared to most of the clustering methods (Jayanti *et al.*, 2009), such a characterization cannot be derived from the similarity of object structure and form. In our approach, two steps can be distinguished. The first step consists in extracting shape distribution characteristics from a set of vertices obtained by discretizing the surfaces of the B-Rep CAD model to be categorized. The second analyses the descriptor values to determine the appropriate category for the considered object.

Our approach can be seen as belonging to the global feature distribution methods (Figure 2). In the context of 3D shape matching, features have a different nature than in classical mechanical engineering (Shah and Mäntylä, 1995), where features correspond to characteristic shape elements which are possibly adopted for part modeling

and associated to some specific meaning (e.g. function, machining operation, assembly operation). For matching, they correspond to geometric and topological properties of 3D shapes whose measuring and comparison allow discriminating objects (Tangelder and Veltkamp, 2008).

Global features characterize the global shape of a 3D model. Examples of these features are the statistical moments of the boundary or the volume of the model, volume-to-surface ratio, or the Fourier transform of the volume or the boundary of the shape. Among the various methods exploiting global features, the one proposed by Corney *et al.* (2002) introduces indices similar to some of those adopted in our work. It uses bounding box-based indices, such as the ratio of the object surface area and the surface area of its convex hull, the percentage of the convex hull volume not occupied by the object and the ratio of the cubed surface area of the hull and the squared volume of the convex hull. To ease the computation, in our work we mainly exploit the oriented bounding box (OBB) as reference instead of the convex hull. The OBB (in opposition to axis-aligned minimum bounding box) is a descriptor introduced by Chang *et al.* (2011). who have also developed methods for its computation. Among them we decided to use the principal components analysis (PCA) because of its easier implementation and its good results for the considered parts. PCA is normally used in all the approaches that require a normalization and alignment between components for their comparison. We do not need the model alignment for our purposes: our aim is not object matching but the classification of objects (possibly with widely varying overall shapes) in groups sharing similar problems with respect to meshing point of view. Therefore, we use the PCA to compute an intrinsic coordinate system for the object.

The global feature distribution method consists in comparing the distribution of global features instead of the global features directly. Among the existing methods, the one proposed by Osada *et al.* (2002) introduces the so-called "D2 Shape Distribution" to represent, in a normalized histogram, the probability of occurrence of Euclidean distances between pairs of randomly chosen points on the skin of the object. The application of shape distributions in the CAD context has been investigated by Cheng *et al.* (2011) and by Ip *et al.* (2002). Ip *et al.* refined Osada's D2 shape distribution function by classifying two random points distances according to whether the joining line segment connecting the points lies both inside and outside the model. Thus, the dissimilarity measure of two objects is a weighted combination of their dissimilarity for the classified D2 distributions. Because of the line classification as inside or outside the model, the method can be applied to volume models, but not to polygonal soups. However, such a distribution only characterizes the overall shape of the object and not the details. Therefore, it is not sufficient to correctly discriminate parts with particular features. Liu *et al.* (2004) propose another variation of a shape distribution function, the thickness histogram estimating thickness of a model from all directions. As explained in Section 5.3, a similar variation of D2 has been adopted, which considers weighted distances between triangles instead of points.

In conclusion, in the proposed approach, we intend to classify objects in categories by defining rules among shape descriptor values. To this aim, advanced shape characterization criteria are built on top of a normalized distance distribution and additional shape descriptors related to the size of the bounding box surrounding the model to be characterized. The approach is simple and easy to implement.

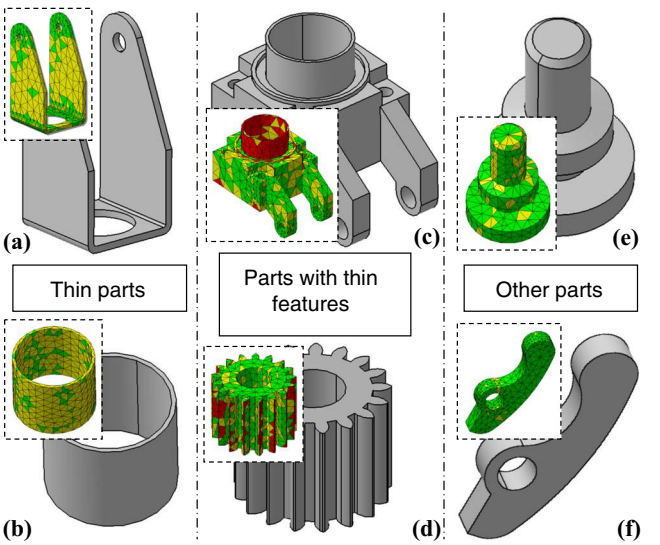
3. Considered categories of objects

To define which categories of objects have to be considered, the different treatments and possible problems that can occur when meshing a CAD model have been analyzed. In particular, three main categories of objects have been identified at first (Figure 3):

- globally thin parts (Figure 3(a) and (b));
- parts containing the so-called thin features, i.e. areas much thinner than the rest of the object (Figure 3(c) and (d)); and
- other parts, i.e. those not belonging to the two previous categories (Figure 3(e) and (f)).

The first and second categories gather CAD models that can be subjected to some idealization processes and/or whose meshing may require manual time-consuming adjustment steps (e.g. local refinement, swap and removal operations). As illustrated in Figures 3(a) and (b), a thin end fitting and a thin tube have been meshed using the default size suggested by the tool thus resulting in bad quality elements (Table I). For the tube, 58.3 percent of the produced tetrahedrons have an aspect ratio smaller than 0.5 which is not satisfactory (Bern and Plassman, 2000). In this case, the engineer has to spend more time adjusting the parameters to improve the quality of the elements.

As illustrated with the body and gear of Figures 3(c) and (d), the second category also includes objects deserving a particular attention for the choice of the mesh size. Here, due to the fact that there are large dispersions of the object's thicknesses, the aspect ratio of some tetrahedrons is even worse than for the first category in these areas which are clearly thinner than the rest of the object (Table I).



Notes: (a, b) Globally thin parts; (c, d) parts with thin features; and (e, f) other parts not belonging to the two first classes

Figure 3.
Examples of parts of the three identified categories

Lastly, today, the last category gathers all the CAD models corresponding neither to globally thin parts nor to parts with thin features, despite the fact that their meshing can be problematic. Moreover, depending on the simulation model that is built at a later stage, some elements with an aspect ratio considered "bad" from the geometric point of view can be "acceptable" from the point of view of accuracy in simulation results. This happens when those badly shaped elements are located in areas where the stress is not important and consequently where the engineer will not focus. In this work we do not exploit the associated simulation-related semantics and we do focus on the shape characteristics of the parts. We plan to refine the classification in the future, thus including also such cases, by combining in the rules global feature information with simulation data.

4. A two-step modular approach

With the purpose of being as independent as possible from both the type of geometric representations (e.g. points cloud, polygons soup, structured meshes, B-Rep models) and adopted CAD modeler, the kernel of the proposed classification works on polygons soups exported in a stereo lithography (STL) file format. This kernel is integrated in a modular two-step approach where shape descriptors are first computed and then used to evaluate shape criteria enabling the classification (Figure 4). The process starts with a unique set of B-Rep models saved in a STEP file format, and ends with three sets of B-Rep models distributed according to the three previously identified categories (Figure 3). In the meantime, each B-Rep models is tessellated face after face and the resulting soup of triangles is saved in a STL file. For each of the STL files, shape descriptors are computed and stored in a TXT file format. The adopted shape descriptors are further developed in Section 5. Then, classification criteria can be evaluated to distribute CAD models according to the three sets: thin parts, parts with thin features and other parts. The adopted classification criteria are presented in Section 6.

STL models are not watertight meshes, i.e. they do not enclose a volume. Thus, the developed algorithm is able to compute shape descriptors on a soup of triangles (polygons). It can also easily deal with many 3D models available over the web. Since most of the models found on the internet are polyhedral models defined in a file format supporting the visual appearance, the proposed approach can be used on a wide variety of data sets.

However, this choice presents some restrictions concerning the type of shape elements on which the analysis can be performed. Most of the current CAD modelers are exploiting a design-by-feature approach for the model creation. While directly reasoning on the part constituting features seems at a first glance suitable when dealing with CAD models, it is not appropriate. First, despite the huge amount of work performed in this direction, there is not a standardized exchange format yet and ad hoc

	Q∈[0,0.3]	Q∈[0.3,0.5] (%)	Q∈[0.5,1] (%)
End fitting (a)	–	27.2	72.8
Tube (b)	–	58.3	41.7
Body (c)	5.4%	27.9	66.7
Gear (d)	8.5%	23.2	68.3
Axis (e)	–	12.8	87.2
Rocker (f)	–	7.8	92.2

Table I. Distribution of the elements according to their aspect ratios within the interval [0,1]

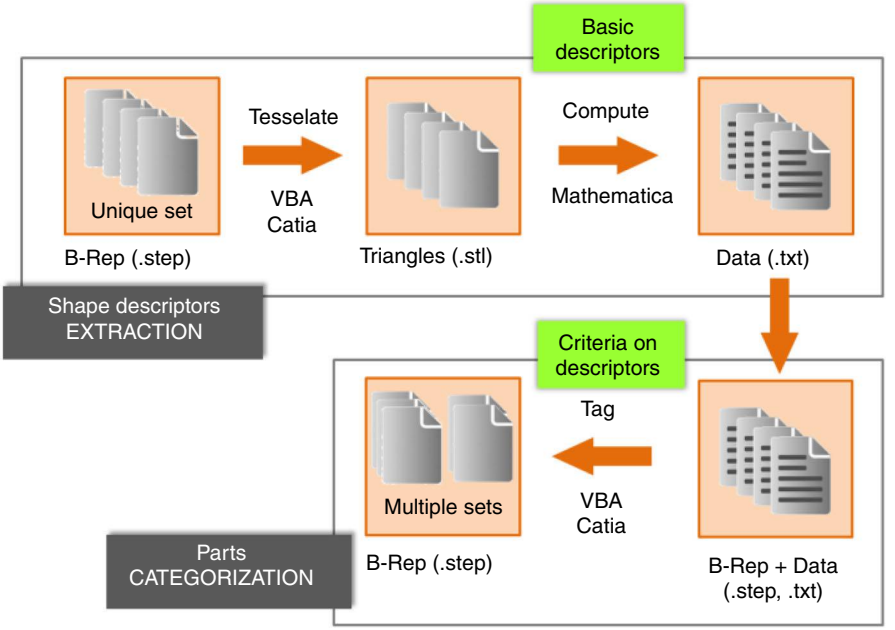


Figure 4.
A two-step modular approach based on the basic descriptors extraction as well as on the evaluation of the criteria related to shape categories

integrations have to be provided for the various considered CAD environments (Shah and Mäntylä, 1995). More important is the impossibility to detect all the possible problematic configurations. Some features cannot be directly visible in the building tree of a CAD model but they can result of their reciprocal position or interaction. Figure 5 illustrates a configuration in which a thin feature results from a set of features but it is not clearly identified as such in the building tree.

5. Adopted shape descriptors

According to the characterizing configurations for the above classes, various existing shape descriptors have been analyzed and the most meaningful have been considered

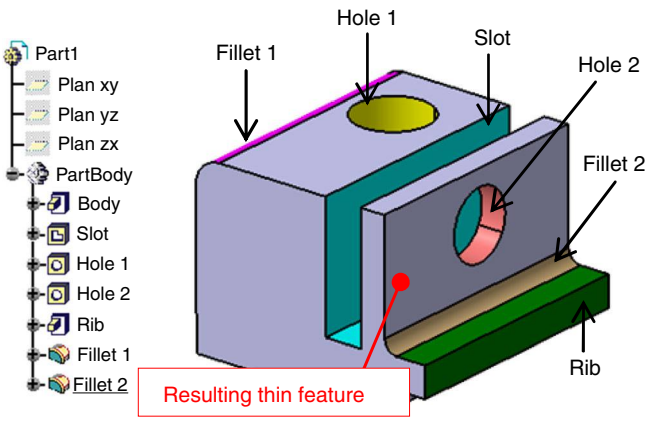


Figure 5.
Example of a thin feature resulting from the association of other features identified in the building tree

and adapted. On one hand, globally thin parts are characterized by having a limited thickness (i.e. small distance between opposite points over the object surface), which results in a largely predominant dimension or in being rather empty internally. On the other hand, parts with thin features are characterized by rather localized areas, which are globally small if compared to the overall object, presenting small distances between opposite points over the object surface. Therefore, the considered descriptors include the area, the volume, the dimensions, and the thicknesses of the parts and associated OBBs.

5.1 Area and volume of the model

In the proposed approach, the area and volume of the object are not considered as standalone shape descriptors since these quantities are definitely linked to the absolute size of the model. However, they can help understanding how much the object is dense or at the opposite made of empty areas if compared to the area and volume of the object bounding box. This aspect is further developed in Section 5.2.

If we go back to the computation of these two basic quantities, the calculation of the model overall area is straightforward and can be obtained by summing up the area of each triangle. As far as volume is concerned, we have adopted an extension of the method used to compute the internal area of a closed planar curve. In 2D, the internal area of a closed curve can be approximated while dividing the oriented bounding curve in several segments $[P_iP_{i+1}]$ (Figure 6). For each oriented segment, an oriented triangle $P_iP_{i+1}O$ is built using the origin O of the reference frame as the third vertex. The signed area of those oriented triangles can be computed using a simple vector product (^) as follows:

$$\mathrm{Area}(P_iP_{i+1}O) = \left(\overrightarrow{P_iP_{i+1}} \wedge \overrightarrow{P_iO}\right).\vec{z}/2 \tag{1}$$

The internal area of the curve is then obtained by summing up the signed areas:

$$\mathrm{Area} = \sum_i \mathrm{Area}(P_iP_{i+1}O) \tag{2}$$

In 3D, the principle is similar. Instead of computing the area of the oriented triangles, we sum up the signed volumes of the oriented tetrahedra. Each tetrahedron is defined with an oriented triangle and the origin of the reference frame as forth vertex.

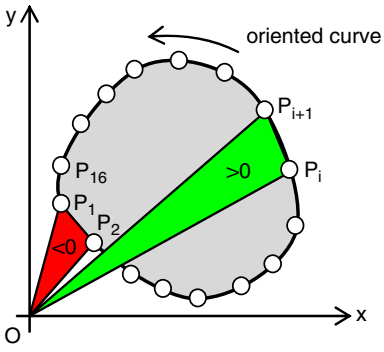


Figure 6. Computation of the area inside a closed curve

5.2 OBB

Computing the minimal OBB of an object consists in finding a rectangular parallelepiped of minimal volume enclosing a set of vertices distributed on the object surface. In our approach, those vertices are directly extracted from the STL file. To get this minimal OBB, we decided to use a famous and basic but efficient method which is the PCA. The PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. In other words, the PCA method means computing the covariance matrix of the set of vertices. Then, the three axes (vectors) of inertia are obtained by computing the eigenvectors of the covariance matrix. The last element is the center of gravity, whose coordinates (X_G, Y_G, Z_G) are the mean of the coordinates (X, Y, Z) of the set of points. As a consequence, the three axis of inertia and the center of gravity of the model represent the coordinate system intrinsic to the object. Then, the three D_{BBi} dimensions of the bounding box are computed as the difference between the maximum and the minimum coordinates of the object points on the three axes. These dimensions are ordered so that $D_{BB1} \geqslant D_{BB2} \geqslant D_{BB3}$.

As previously stated, it would not be meaningful to build classification criteria on top of shape descriptors that would use absolute basic quantities like area or volume. Hence, the computation of the minimal OBB is used as a mean to evaluate how much the object is filled or rather empty with respect to its bounding box. As a consequence, the following ratio k_v is introduced as a shape descriptor used to build the classification criteria (Section 6) at a later stage:

$$k_V = \frac{V_{model}}{V_{BB}} \tag{3}$$

Also, the ratios among the D_{BBi} dimensions reveal if the object OBB is more like a thin plate or like a cuboid. Actually, plate-like shapes are characterized by having one rather small out of the three dimensions if compared to the others. Therefore, the two following ratios will also be used as shape descriptors:

$$k_{D_1} = \frac{D_{BB1}}{D_{BB2}} \text{ and } k_{D_2} = \frac{D_{BB2}}{D_{BB3}} \tag{4}$$

Since $D_{BB1}/D_{BB3} = k_{D_1}.k_{D_2}$, adding another independent condition on the ratio D_{BB1}/D_{BB3} would lead to conflicting configurations. The condition on D_{BB1}/D_{BB3} can be derived from the conditions on k_{D_1} and k_{D_2} and for this reason it has not been considered.

Lastly, the diagonal of the OBB is used to normalize the other descriptors in order to have a relative quantification of the thickness with respect to the actual product size:

$$\text{diag}_{BB} = \sqrt{\sum_{i=1}^{3} D_{BBi}{}^2} \tag{5}$$

5.3 Distance distribution

One of the main descriptors used in our work is the so-called distance distribution, inspired from the work of Osada *et al.* (2002) on the D2 shape distribution descriptor. Osada *et al.* use the distance distribution to characterize the overall shape of the object and discriminate objects with different gross shapes. It is computed by measuring the

distance between points sampled over the surface. In order to be independent from the model tessellation, points are selected through stochastic methods while taking into consideration the cumulative area of the triangles around them. In our work, the aim is to recognize thin areas on either subparts of the shape or on the entire shape. In this perspective, we have adapted the idea of distance computation while adding some conditions to compute the thickness of the part. More precisely, the main differences with the D2 descriptor are related to the fact that we compute distances between triangles and not vertices (Section 5.3.1), and that some additional conditions are put in place to guarantee the independency from the underlying geometric representations (Section 5.3.2). Since we are dealing with tessellated CAD models, i.e. with triangles having widely varying dimensions and shapes, the stochastic selection used by Osada *et al.* has not been adopted in our implementation and the computations are performed on all the triangles.

5.3.1 Distance between triangles. While the distance between two parallel triangles T_i and T_j is a clear and easy way to compute concept, this notion has to be clarified when considering non-parallel faces. Hence, being $\mathbf{n}_i$ and $\mathbf{n}_j$ the normals of T_i and T_j, the developed algorithm acts differently according to the cases for which it is considered meaningful to compute the distances:

If [$\mathbf{n}_i.\mathbf{n}_j \leqslant \cos(170°)$] **Then**
  T_i and T_j are considered as "almost" parallel;
Else If [$\cos(170°) < n_i.\mathbf{n}_j < \cos(120°)$] **Then**
  T_i and T_j are considered as "potentially" being in a teeth-like configuration;
Else
  T_i and T_j are not in a meaningful configuration.
End If

As a matter of fact, the first test identifies configurations where the angle between the two faces is smaller than 10°. This threshold has been adjusted heuristically to prevent badly identified configurations notably on noisy data. Combined with other tests, the second thresholds have been tuned to clearly identify teeth-like configurations. This may happen when the angle between the two faces is in between 10° and 60°. In the future, one could also imagine that these thresholds are adapted according to the nature of the original geometric models (e.g. B-Rep model, scanned data) and accuracy of tessellation.

Therefore, two main configurations have to be distinguished:

- When two triangles are identified as "almost" parallel, we have to check if they are facing each other and if their normal vectors do not point toward one another. These additional verifications help avoiding irrelevant distance computations. If these verifications were to fail, i.e. if the triangles are not facing each other and/or if the normal vectors point toward one another, it means that the triangles enclose a non-meaningful volume. The devised method performs both verifications simultaneously in a single test. It consists in projecting the three vertices $\mathbf{V}_{i1}$, $\mathbf{V}_{i2}$, $\mathbf{V}_{i3}$ and the center of gravity $\mathbf{G}_i$ of the first triangle T_i into the other one T_j. If the projection $\mathbf{G}_i$ lies inside T_j, like in Figure 7(a), we consider that the distance computation is meaningful for the thickness distribution. Moreover, by using the opposite of the normal vector for projecting $\mathbf{G}_i$ in T_j, the algorithm allows the differentiation between parallel faces enclosing the object material or

the void. Indeed, we do not care of the distance between parallel faces not enclosing the object material. Being $\mathbf{P}_i$ the projection of $\mathbf{G}_i$ in T_j (Figure 7(a)), this additional test distinguishes two configurations:

If $[\mathbf{G}_i\mathbf{P}_i.\mathbf{n}_i < 0]$ **Then**
  T_i and T_j enclose a non-empty space;
Else
  T_i and T_j enclose an empty space and are not considered for the distance computation anymore.
End If

In case point $\mathbf{P}_i$ is projected outside T_j, the algorithm checks if at least the projection of one of the three vertices of T_i is inside T_j (Figure 7(b)). There is a case which is not taken into account, that-is-to say when two faces are facing but all the projections (center of gravity and other points) are not lying in the other face. This is not a critical issue because we assume that if a triangle T_i is in this case with a triangle T_j, then T_i (or T_j) is almost parallel to a triangle adjacent to T_j (resp. T_i).As our last step, if all the tests are validated, the distance between T_i and T_j is evaluated as follows:

$$d_{ij}^{\text{parallel}} = \mathbf{G}_j\mathbf{G}_i.\mathbf{n_i} \tag{6}$$

The area of the two triangles is also stored so that the distance distribution can be normalized and decoupled from the underlying tessellation:

$$A_{ij} = \text{area}(T_i) + \text{area}(T_j) \tag{7}$$

- When two faces are considered as "potentially" being in a teeth-like configuration, the distance between the two triangles is considered to be the smallest between the three points of each faces:

$$d_{ij}^{\text{teeth}} = \min\left[d\left(V_{ip}, V_{jq}\right), (p, q) \in \{1..3\}^2\right] \tag{8}$$

The area of the two triangles is also stored with the same equation as (7).

Finally, the overall algorithm, made of two imbricated loops, reviews each triangle T_i and carries out the above tests with the other triangles T_j. When one of the two configurations is identified, the corresponding distance and area are inserted in a distance and configuration list.

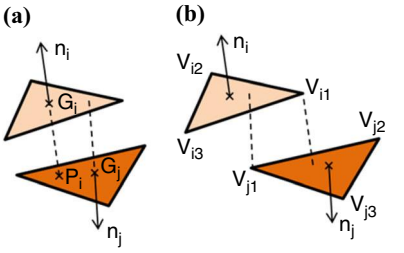


Figure 7. Parallel faces identification with projection of the gravity center of the triangle (a); and projection of the face vertices (b)

5.3.2 Normalization of the distribution. To get the final distance distribution normalized and tessellation-invariant, we have to weight the distribution of the d_{ij} distances using ratios which characterize the contribution of the A_{ij} areas with respect to the overall object area. In this way it is possible to know which percentage of the object is characterized by the same thickness. As a last step, to make the distances meaningful if compared with the dimensions of the whole object, and thus make comparable objects of very different dimensions, we have to normalize the d_{ij} values by dividing them by the diagonal $diag_{BB}$ of the minimal bounding box (Equation (5)). In the end, we obtain a normalized distance distribution used as a shape descriptor that is then combined with the previously introduced descriptors to define classification criteria (Section 6).

Figure 8 shows two examples of parts with the associated thickness distribution function and the filling percentage of their respective OBB obtained by the ratio between the volumes of the object and of its bounding box (Equation (3)). For the U-like

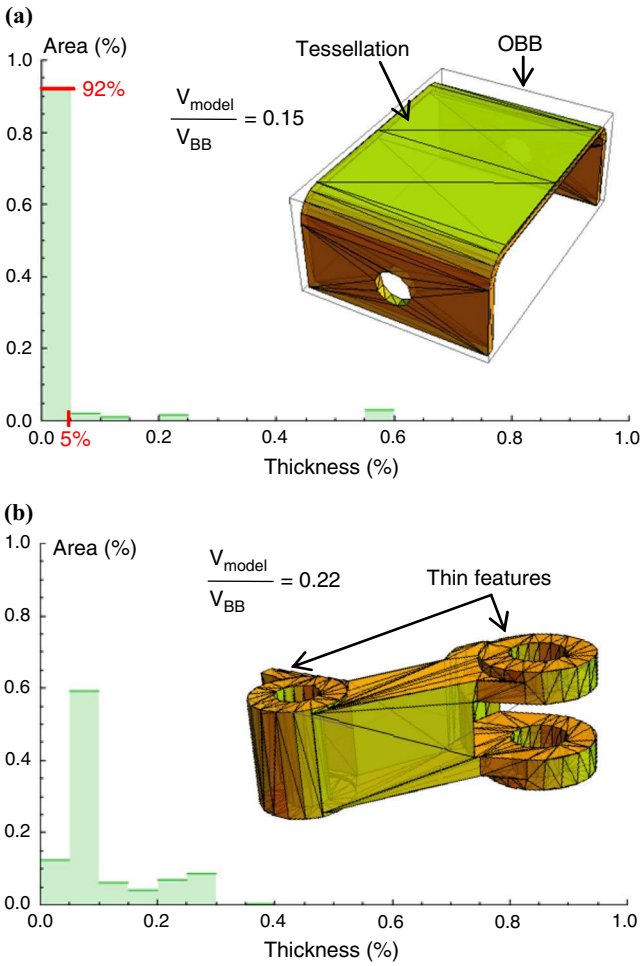


Figure 8. Normalized distance distributions on a globally thin part (a); and on a part with thin features (b)

plate with the two holes (Figure 8(a)), the ratio between the volumes of the model and associated OBB is equal to 0.15 which means that 15 percent of the OBB is filled in. The thickness distribution reveals one main thickness inside the model, since 92 percent of the total model area is associated with thickness between 0 and 5 percent of the bounding box diagonal $diag_{BB}$. The x-axis indicates the value of the normalized thickness while the y-axis represents the associated area percentage. For the end fitting example (Figure 8(b)), the distance distribution does not reveal a clear main distance. According to the adopted criteria described in Section 6, the first example will be classified as a thin part whereas the second part will be classified as a part with thin features.

6. Part classification

This section addresses the way in which the afore-mentioned descriptors can be combined to define criteria used to categorize the objects according to the classification introduced in Section 3. These criteria use some thresholds which have been empirically tuned (Section 6.3).

In addition to the notations introduced in Section 5, we must define the following function used to synthetize and analyze the distance distribution:

Function z = F[x; y]

i.e. function of the distance distribution so that z represents the percent of the total area of the model associated to distances between x and y percent of the diagonal of the bounding box.

For example, "**F**[0; 0.25] = 0.88" means that "88 percent of the total surface is associated to a thickness between 0 and 25 percent of the diagonal of the bounding box."

6.1 Thin parts

The thin part class includes different types of CAD models, such as those similar to thin plates, or having an arbitrary shape with almost constant thickness distribution, or presenting a large emptiness of the bounding box (Figure 9). Different criteria are used to distinguish the typology of thin parts, as described in the following of this section. It is clear how some thin parts can satisfy more than one criterion simultaneously.

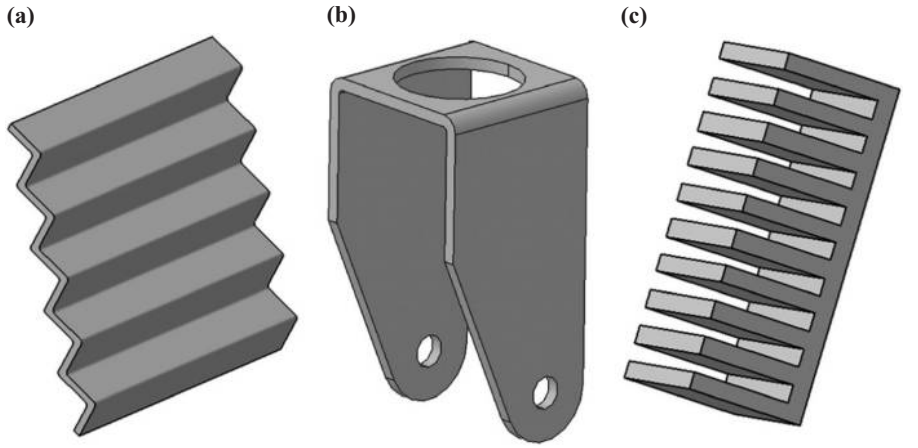


Figure 9. Examples of thin objects

The first criterion (crit.1 in Table II) identifies plates, based on the fact that plate-like objects have one smaller dimension of the OBB out of three, with one part filling at least the half of its bounding box. This criterion allows the identification of thin parts (Figure 9(a)).

The second criterion (crit.2 in Table II) aims at identifying thin parts having simultaneously two similar OBB dimensions while the third dimension is rather small; they also have a rather important filling of the OBB. Therefore, part in Figures 9(a) is kept by this criterion in the thin part class.

The third criterion (crit.3 in Table II) considers objects having a general overall shape with no predominant dimension of the OBB. This kind of objects is characterized by the fact that the volume occupied by the part in the enclosing OBB is just a small part (ratio $V_M/V_{BB} < 20$ percent). The aim is to include parts which are really extended in their bounding box, but with no volume, as the parts in Figures 9(b) and 10(a). Indeed, the difference between the object in Figure 10(a) and the one in Figure 10(b) is that the first has a rather small volume compared to the volume of its OBB and has a quite constant thickness, while the second has some thin features which have been

Table II. List of the criteria and thresholds adopted for the thin parts class

	BB filling	BB size rate	Distance distribution
Crit.1	$V_M/V_{BB} > 0.5$	$D_{BB2}/D_{BB3} > 6$	
Crit.2	$V_M/V_{BB} < 0.8$	$D_{BB1}/D_{BB2} < 1.75$ $D_{BB2}/D_{BB3} > 6$	
Crit.3	$V_M/V_{BB} < 0.2$		
Crit.4	$V_M/V_{BB} > 0.2$		$F[0;0.05] > 0.6$

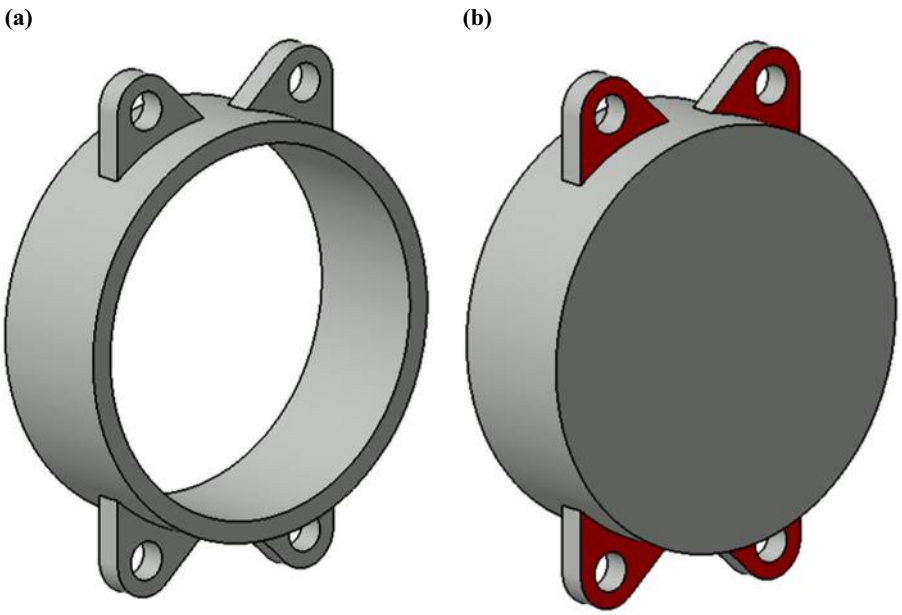


Figure 10. Examples of a thin part (a); and of a part with thin features (red) having a similar overall shape (b)

highlighted in red. With this criterion, we can exclude the second part (Figure 10(b)) from the thin part class.

Finally, the fourth criterion (crit.4 in Table II) considers the distribution descriptor and identifies thin parts whose volume is filling more than 20 percent of its bounding box volume. Here, the adopted formula is $\mathbf{F}[0;0.25] > 0.6$ which allows us to classify objects like the one depicted in Figure 9(c). It identifies a model that has a little filling of its bounding box but more than 60 percent of the total surface is associated to thickness under 25 percent of the diagonal of the bounding box.

Thin parts are not necessarily difficult to mesh, but most of the time, a designer will have to be careful with these parts.

6.2 Parts with thin features

This class includes all parts containing one or more features which may deserve some manual checks before they are meshed. Indeed, if the automatically proposed meshing size is too large, these features might be badly meshed, giving rise to incorrect simulations.

Parts having thin features are characterized by areas having a rather small thickness if compared to the overall object thickness. Therefore, they are recognized by mainly analyzing the distance distribution. Actually, when the part is not thin in its entirety, the distance distribution detects a local thickness with respect to the rest of the part.

Table III summarizes the defined criteria and thresholds used to detect parts with thin features. The fifth and sixth criteria combine conditions on the volume with conditions on the thickness values. The condition on the volume allows avoiding totally thin parts. The examples of objects classified according to crit.5 and crit.6 of Table III are, respectively, given in Figures 11(a) and (b).

Finally, crit.7 is used to detect parts with thin features defined by non-parallel (//) faces, as for gearwheel teeth (Figure 3(d)), or corresponding to thin elements bounded by non-planar faces, e.g. cylindrical. Here, there is just a value of all-or-nothing and no distribution is compared. So, when faces with an angle between 10° and 60° (Section 5.3.1) have a distance between them under 5 percent of the bounding box diagonal, the part is considered as a part with thin features.

6.3 Adopted thresholds

The thresholds adopted in the proposed criteria have been tuned empirically after several evaluations on multiple test cases. Actually, being the key parameters identified, several verifications have been carried out to tune the thresholds with respect to the aspect ratio of the elements resulting from a meshing with a default size parameter. Thus, it results that the meshing of the parts categorized as either thin part or as part with thin features generates elements with an aspect ratio (Bern and

Table III. List of the criteria and thresholds adopted for the parts with thin features

	BB filling	BB size rate	Distance distribution
Crit.5	$V_M/V_{BB} > 0.2$		$F[0;0.05] > 0$ $F[0;0.05] < 0.6$
Crit.6	$V_M/V_{BB} > 0.5$		$F[0.05;0.25] > 0$ $F[0.3;0.7] > 0.2$
Crit.7			$\min(d_i \text{ between non // faces})/diag_{BB} < 0.05$

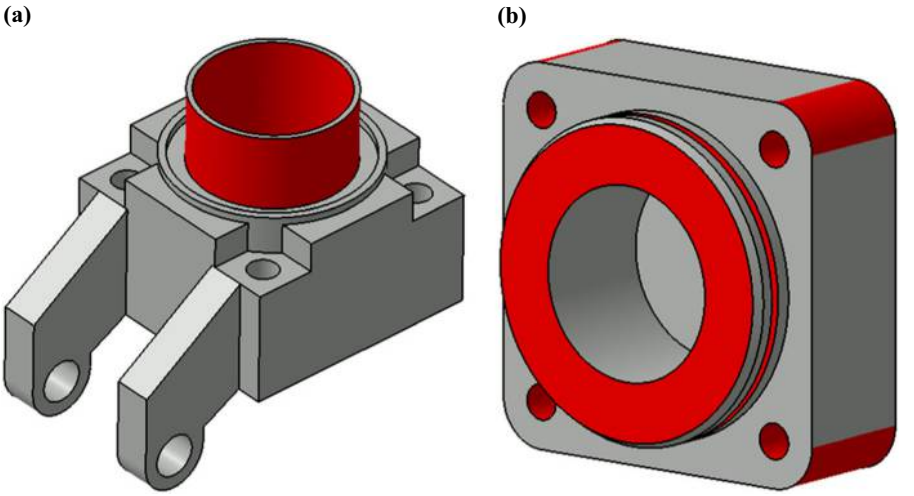


Figure 11. Examples of parts with thin features

Plassman, 2000) smaller than 0.5. As previously stated, this means that the user will have to spend more time adjusting the mesh (Section 1).

In the future, we intend to use machine learning techniques to adjust the values of those thresholds using a wider set of test cases. This is further discussed in the conclusions.

7. Developed prototype and results

The object classification algorithm has been implemented as a VBA macro within CATIA V5. As introduced in Section 4 (Figure 4), the macro reads a STEP file, exports the corresponding tessellation model in a STL file and calls Worlfram Mathematica (Worlfram Research) to compute the proposed shape descriptors. Then, the macro associates to the part a so-called "info file" in which the values of the shape descriptors are stored together with the identified class. These are the data obtained:

- ratio V_M/V_{BB};
- dimensions of the OBB D_{BB1}, D_{BB2} and D_{BB3};
- F[0; 0.05], F[0; 0.25] and F[0.3; 0.7] which are three samples of the distribution function which are used in the categorization;
- indicator of existence of teeth-like thin features (0 or 1); and
- identified class: thin part, part with thin features or other part.

In this version, the info file has not yet been integrated in a PDM system and is stored in the directory where the CAD model is placed. The storage of this information allows a fast reclassification of the parts in case different thresholds are to be considered because they are more suitable for the type of meshing or simulations that need to be carried out. As a matter of fact, even if the chosen threshold values have been demonstrated to give quite good results on the set of considered parts, it is clear that these thresholds might change according to the type of system used for meshing. Therefore, some machine learning techniques would be very useful to adjust the

thresholds to the specific engineering environments. Figure 12 shows the sequence of communication between CATIA V5 and Mathematica according to the framework introduced in Section 4.

The provided macro is automatically activated or on user demand, on either a single model or on a set of models. We actually foresee the possibility of automatically categorizing a part when it is designed, while storing the computed info file as an

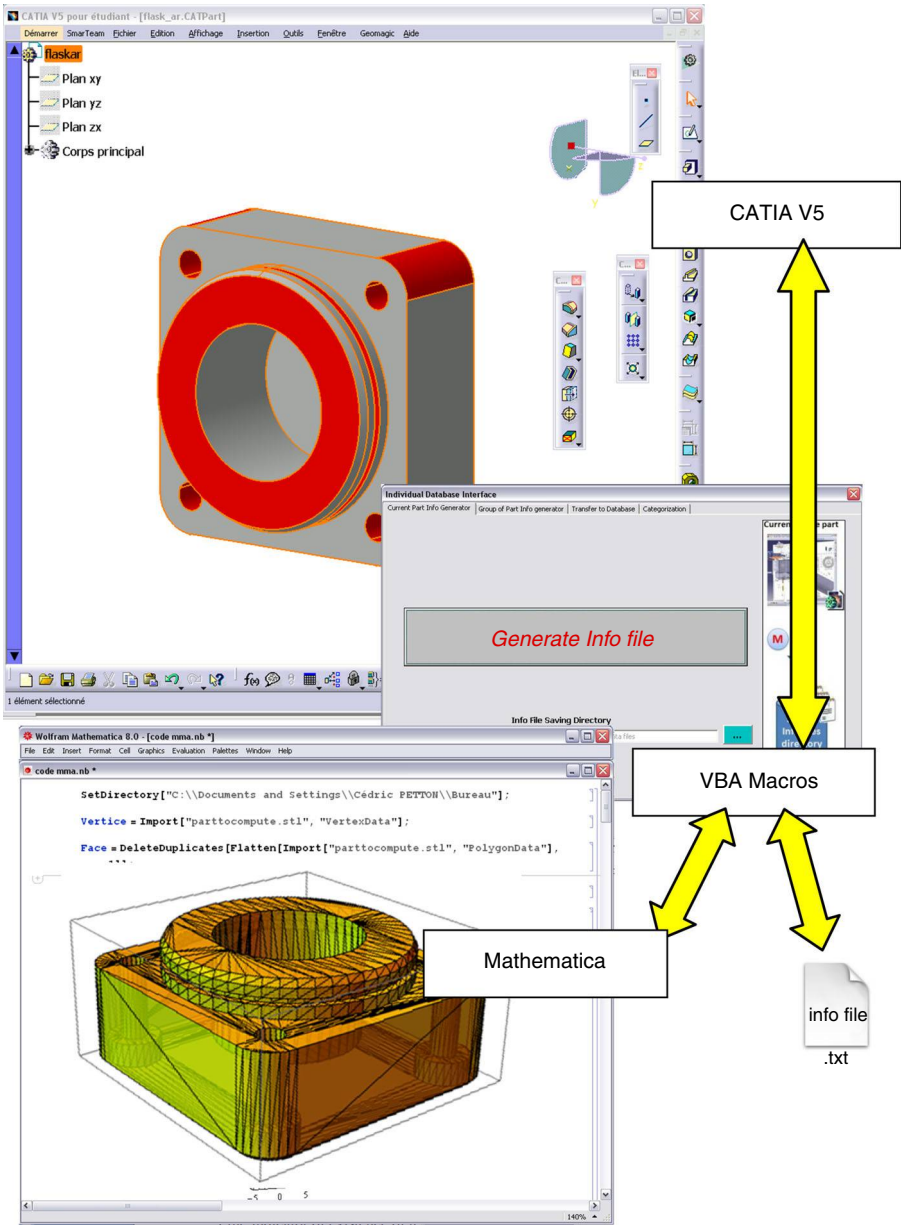


Figure 12.
Sequence of communication between CATIA V5 and Mathematica through VBA Macros

accompanying document of each model. The computed category information can then be used to make the simulation expert immediately aware of potential problems with the part and for supporting him to retrieve similar situations.

The proposed classification approach has been validated on numerous examples. Due to space limits, only seven parts for each category have been depicted in Figure 13 and the values of the shape descriptors have been gathered together in Table IV. The three samples of the distance distribution F[0;0.05], F[0;0.25] and F[0.3;0.7] help understanding the evolution of the underlying distance distribution that is not displayed. The proposed approach works well. The thin parts as well as the parts with thin features have been identified properly. For those parts, the meshing with a default size parameter generates mesh elements with a bad aspect ratio. It means that for those parts, the engineers will have to spend more time during the meshing phase. This can be said also for the parts for which thin features are adjacent to thicker areas. When a part does not satisfy any of the seven criteria, it is classed in the category called "other parts." Lastly, for a given class, one can notice that a part can satisfy several criteria.

8. Conclusions and future work

Anticipating possible problems occurring at different stages of the development process is quite important from the perspectives of both cost saving and finding the achieved solution. The work presented in this paper is a first step toward the definition of a complete toolbox for the classification of parts which are potentially complex to mesh or not, i.e. parts that may generate elements with a bad aspect ratio penalizing the simulation at a later stage. In this perspective, parts are classified depending on their shapes and global features. In this way, designers can anticipate the time they will have to spend on the meshing phases. The classification is based on a set of criteria exploiting several shape descriptors that have been normalized to get rid of the scale of the objects. A specific distance distribution is notably computed to show how the thickness evolves over the entire model.

The kernel of the classification algorithm works on STL files, thus it is made system independent and it can be easily integrated on any CAD systems. In addition, the use of the info file storing all the key shape descriptors and thresholds may allow a quicker and more different categorization of the parts according to the user/system needs, being the most time-consuming activity related to the shape descriptor evaluation. To be fully usable inside companies, the method should be extended to cover the classification of assemblies and accommodate the evaluated descriptors values in a PDM repository. As a matter of fact, in the actual version, assemblies can be either analyzed singularly or they can be analyzed while merging all of them in a new part that is then classified. In the future, we are planning to create a process to automatically perform the required operations to classify existing repositories including both single parts and assemblies.

Current works include the optimization of the algorithm to improve the computation times. As previously stated, we plan to improve the classification and criteria by coupling the shape characteristics and simulation semantics, e.g. loads and blocked displacements. We are also working on new functionalities to assist designers in modifying the various parts categorized as "difficult to mesh" so that they can be meshed. Once again, the idea is to find the right criteria and associated thresholds to adapt the CAD models using, for example, a set of defeaturing operations. Lastly, we foresee the use of machine learning techniques for identifying both the best thresholds and combination of descriptor/simulation data.

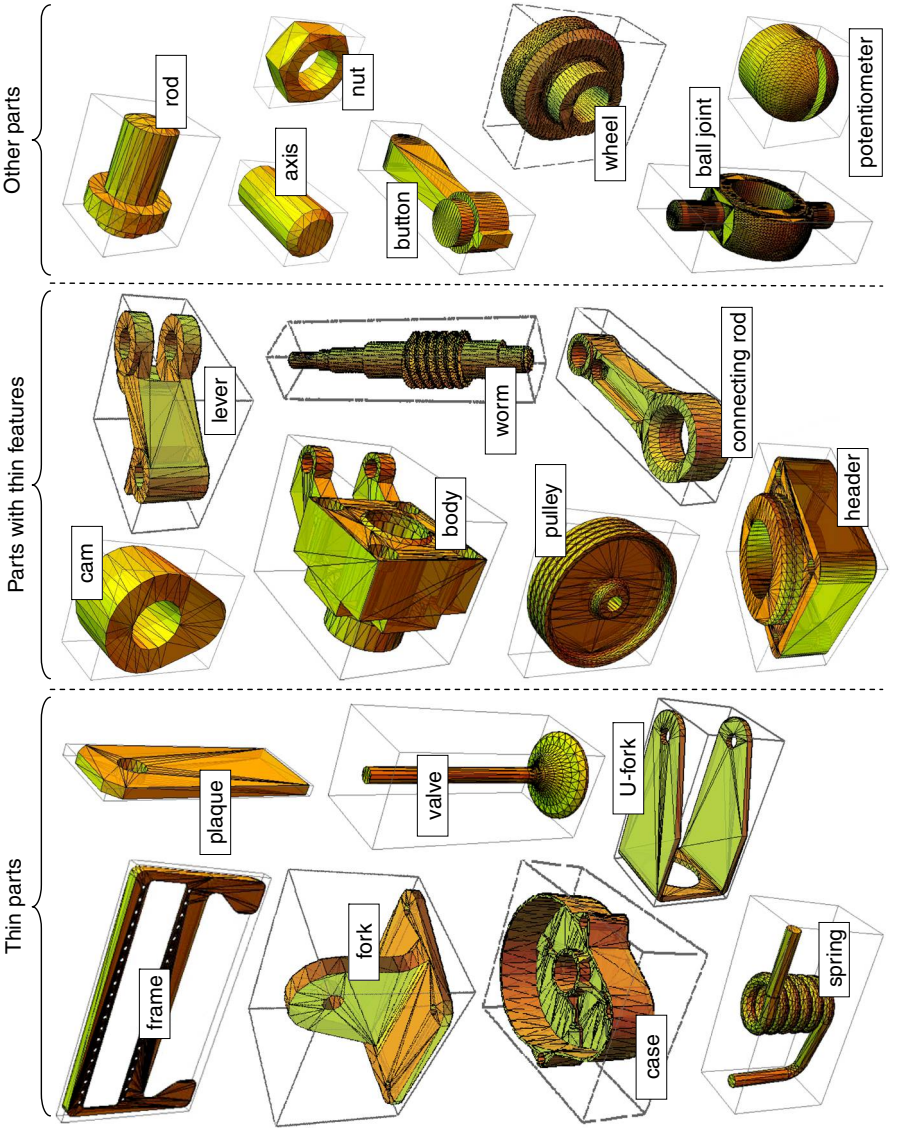


Figure 13.
Categorization of thin parts, parts with thin features and other parts

Table IV. Some figures relative to the shape descriptors and evaluated criteria for the objects in Figure 13

	V_{model}/V_{BB}	D_{BB1}/D_{BB2}	D_{BB2}/D_{BB3}	F[0;0.05] (%)	F[0;0.25] (%)	F[0.3;0.7] (%)	Criteria
Thin parts							
Frame	0.08	1.7	13.4	90.7	92.7	0.0	2 and 3 and 7
Plaque	0.95	2.9	7.0	81.6	97.9	0.0	1 and 4
Fork	0.12	1.3	1.1	58.3	88.1	3.4	3
Valve	0.06	2.5	1.0	0.0	35.4	3.6	3 and 7
Case	0.09	1.2	2.3	90.6	98.5	0.1	3 and 7
U-fork	0.13	2.0	1.0	88.3	96.5	1.9	3 and 7
Spring	0.10	1.8	1.4	10.3	80.5	1.5	3 and 7
Parts with thin features							
Cam	0.60	1.3	1.7	0.0	52.7	35.8	6
Lever	0.22	1.0	2.3	12.4	89.1	0.3	5 and 7
Body	0.26	1.2	1.2	53.9	89.0	0.9	5 and 7
Worm	0.30	4.6	1.0	1.5	61.7	2.1	5 and 7
Pulley	0.44	1.0	4.0	32.5	65.5	16.6	5 and 7
Connecting rod	0.32	2.9	2.1	36.1	81.3	1.8	5 and 7
Header	0.68	1.0	2.1	36.1	90.8	5.1	5 and 7
Other parts							
Rod	0.36	1.4	1.0	0.0	43.4	50.3	none
Axis	0.75	2.1	1.0	0.0	0.0	76.0	none
Nut	0.48	1.2	2.1	0.0	52.8	14.5	none
Button	0.38	3.1	1.1	0.0	81.0	0.0	none
Wheel	0.44	1.0	1.8	0.0	54.3	8.5	none
Potentiometer	0.55	1.0	1.1	0.0	45.9	12.1	none
Ball joint	0.32	1.7	2.4	0.0	73.8	22.7	none

References

Bai, J., Gao, S., Tang, W., Liu, Y. and Guo, S. (2010), "Design reuse oriented partial retrieval of CAD models", *Computer-Aided Design*, Vol. 42 No. 12, pp. 1069-1084.

Bern, M. and Plassman, P. (2000), "Mesh generation", in Sack, J.-R. and Urrutia, J. (Eds), *Handbook of Computational Geometry*, Elsevier Science Publishers, B. V. North-Holland, Amsterdam, pp. 291-332.

Bespalov, D., Ip, C.Y., Regli, W.C. and Shaffer, J. (2005), "Benchmarking CAD search techniques", *Proceeding of ACM Symposium on Solid and Physical Modeling*, pp. 275-286.

Brière-Côté, A., Rivest, L. and Maranzana, R. (2012), "Comparing 3D CAD models: uses, methods, tools and perspectives", *Computer-Aided Design and Applications*, Vol. 9 No. 6, pp. 771-794.

Cardone, A., Gupta, S.K. and Karnik, M. (2003), "A survey of shape similarity assessment algorithms for product design and manufacturing applications", *Journal of Computing and Information Science in Engineering*, Vol. 3 No. 2, pp. 109-118.

Chang, C.-T., Gorissen, B. and Melchior, S. (2011), "Fast oriented bounding box optimization on the rotation group SO(3,R)", *ACM Transactions on Graphics*, Vol. 30 No. 5, article no. 122.

Cheng, H.-C., Lo, C.-H., Chu, C.-H. and Kim, Y.S. (2011), "Shape similarity measurement for 3D mechanical part using D2 shape distribution and negative feature decomposition", *Computers in Industry*, Vol. 62 No. 3, pp. 269-280.

Chu, H. and Hsu, Y.-C. (2006), "Similarity assessment of 3D mechanical components for design reuse", *Robotics and Computer-Integrated Manufacturing*, Vol. 22 No. 4, pp. 332-341.

Corney, J., Rea, H., Clark, D., Pritchard, J., Breaks, M. and Macleod, R. (2002), "Coarse filters for shape matching", *Computer Graphics and Applications*, Vol. 22 No. 3, pp. 65-74.

Cuillière, J.-C., François, V., Souaissa, K., Benamara, A. and BelHadjSalah, H. (2011), "Automatic comparison and remeshing applied to CAD model modification", *Computer-Aided Design*, Vol. 43 No. 12, pp. 1545-1560.

Demirci, M.F., van Leuken, R.H. and Veltkamp, R.C. (2008), "Indexing through laplacian spectra", *Computer Vision and Image Understanding*, Vol. 110 No. 3, pp. 312-325.

Hong, T., Lee, K. and Kim, S. (2006), "Similarity comparison of mechanical parts to reuse existing designs", *Computer-Aided Design*, Vol. 38 No. 9, pp. 973-984.

Ip, C.Y., Lapadat, D., Sieger, L. and Regli, W.C. (2002), "Using shape distributions to compare solid models", *Proceedings of ACM Symposium on Solid and Physical Modeling*, pp. 273-280.

Iyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y. and Ramani, K. (2005), "Three-dimensional shape searching: state-of-the-art review and future trends", *Computer-Aided Design*, Vol. 37 No. 5, pp. 509-530.

Jayanti, S., Kalyanaraman, Y. and Ramani, K. (2009), "Shape-based clustering for 3D CAD objects: a comparative study of effectiveness", *Computer-Aided Design*, Vol. 41 No. 12, pp. 999-1007.

Li, M., Zhang, Y.F. and Fuh, J.Y.H. (2010), "Retrieving reusable 3D CAD models using knowledge-driven dependency graph partitioning", *Computer-Aided Design and Applications*, Vol. 7 No. 3, pp. 417-430.

Li, M., Zhang, Y.F., Fuh, J.Y.H. and Qiu, Z.M. (2011), "Design reusability assessment for effective CAD model retrieval and reuse", *Int. Journal Computer Applications in Technology*, Vol. 40 Nos 1/2, pp. 3-12.

Liu, Y., Pu, J., Zha, H., Liu, W. and Uehara, Y. (2004), "Thickness histogram and statistical harmonic representation for 3D model retrieval", *Proceeding of 3D Data Processing, Visualization and Transmission*, pp. 896-903.

Msaaf, O., Maranzana, R. and Rivest, L. (2007), "Part data mining for information re-use in a PLM context", Paper No. GT2007-27966, *Proceedings of ASME Turbo Expo 2007, Montreal*, May 14-17.

Ohbuchi, R., Minamitani, T. and Takei, T. (2005), "Shape-similarity search of 3D models by using enhanced shape functions", *Int. Journal of Computer Applications in Technology*, Vol. 23 Nos 2/4, pp. 70-85.

Osada, R., Funkhouser, T., Chazelle, B. and Dobkin, D. (2002), "Shape distributions", *ACM Transactions on Graphics*, Vol. 21 No. 4, pp. 807-832.

Paquet, E., Murching, A., Naveen, T., Tabatabai, A. and Rioux, M. (2000), "Description of shape information for 2-D and 3-D objects", *Signal Processing: Image Communication*, Vol. 16 No. 1, pp. 103-122.

Pernot, J.-P., Giannini, F. and Petton, C. (2012), "Categorization of CAD models based on thin part identification", *Proceedings of TMCE'12, Karlsruhe*, May 7-11.

Shah, J.J. and Mäntylä, M. (1995), *Parametric and Feature-based CAD/CAM*, John Wiley & Sons Inc., New York, NY.

Tangelder, J. and Veltkamp, R. (2008), "A survey of content based 3D shape retrieval methods", *Multimedia Tools and Applications*, Vol. 39 No. 3, pp. 441-471.

Worlfram Research, "Mathematica software", available at: www.wolfram.com (accessed June 2014).

Zhu, K., San Wong, Y., Tong Loh, H. and Feng Lu, W. (2012), "3D CAD model retrieval with perturbed laplacian spectra", *Computers in Industry*, Vol. 63 No. 1, pp. 1-11.

About the authors

Jean-Philippe Pernot is a Full Professor of Mechanical Engineering at the Arts et Métiers ParisTech. Former student from the Ecole Normale Supérieure de Cachan (1997-2000), he succeeded the Agrégation of Mechanics in 2000 and received his PhD in 2004 from both the Institut National Polytechnique of Grenoble (France) and the Università degli Studi di Genova (Italy). His thesis has been awarded in 2005. He is a Laureate of several scientific and best paper awards. He is also beneficiary of a Scientific Excellence Grant. He is author of more than 60 international scientific papers. He is member of several international program committees and editorial boards. His works focus on the specification and development of semantic-oriented models, methods and tools for the generation and treatment of Digital Mock-Up in aesthetic and engineering designs. More precisely, his research activities focus on geometric modeling, free-form feature modeling, direct finite element meshes modification, deformation techniques, polyhedral simplification, use of images for geometric model processing, reverse engineering, virtual reality. Professor Jean-Philippe Pernot is the corresponding author and can be contacted at: jean-philippe.pernot@ensam.eu

Dr Franca Giannini is a Senior Researcher at the IMATI-CNR in Genova, Italy. She graduated in Applied Mathematics from the University in Genoa in November 1986. Since then, taking into account the evolution of the available technologies and changes in working processes, she concentrated on different issues for the specification of tools and methodologies for 3D geometric model representation, analysis and synthesis. In particular, she focusses on the development of shape processing and modeling tools adaptable to the application needs by exploiting contextual knowledge. She has participated and been responsible for IMATI of several national and international projects. She is co-author of two patented software for automatic feature recognition for hybrid solid representation. The results of her research activity have been published in more than 100 reviewed papers presented in international conferences and journals. Her current research interests include multidimensional media modeling and understanding and related knowledge formalization in applications contexts.

Cédric Petton is a Student of Arts et Métiers ParisTech working in the field of computer-aided design and more precisely on the way CAD models can be categorized. He has developed various skills in the processing of geometric models and in the identification intrinsic and extrinsic parameters for CAD models classification.