Feature interference in free form template matching

Y. Song, J. S. M. Vergeest and I. Horváth

Faculty of Design, Engineering and Production, Delft University of Technology, 2628CE, Delft, The Netherlands
y.song@io.tudelft.nl j.s.m.vergeest@io.tudelft.nl i.horvath@io.tudelft.nl

Abstract

Template matching is a useful method in the reconstruction of free form surfaces. In this paper, matching of free form surfaces with parameterized templates is studied with the emphasis on feature interference. By their definitions in 3-Dimensional space, free form features are introduced, analyzed and parameterized first. An optimization function based on a Hausdorff-like shape distance measuring method is proposed and applied as measuring method between the digitalized model surface and the parameterized feature template. The proposed method not only can fit isolated free form features, but also works when a feature is interfered by other free form features in some specified conditions. A series of numerical experiments have been conducted to demonstrate the proposed methods based on ACIS® and Open Inventor®. Fitting strategies are proposed as well.

1. Introduction

Precedent designs can be reused by reverse engineering. Reverse engineering is the process of obtaining a geometric Computer Aided Design (CAD) model from measurements of existing artifacts[1]. The purpose of reverse engineering can be either to provide digital support for subsequent life cycle stages of a product for which no CAD model is available, or to support the redesign of an existing product. In the past decades, although reverse engineering is a well established technology in many areas, its application for design is still an unsolved issue when redesigning goes beyond the adjustment of the originally defined design parameters especially in free form surfaces. The problem in this issue is the non-uniqueness of the types of parameters for a given object[2]. Generally, designers tend to reason about and operate on higher level entities than geometric constituents such as points, curves and individual surfaces. But currently, if the designers want to modify a reconstructed free form surface, they always face the problem of dealing with many control points with some un-predictable methods. For solving this problem, the feature concept has been introduced to free form surface reverse engineering.

As a key element in shape modeling, the feature gains enough attention after it was introduced. Generally, a feature means the generic shapes or characteristics of a product with which engineers can associate certain attributes and knowledge useful for reasoning about the product[3]. A feature offers the advantage of treating sets of elements as a single entity. Considering alternative solutions and shortening the time required for models changes, it is quite clear that using features as design primitives improves the efficiency in creating the product model[4].

While the concept of feature has been mainly investigated in the mechanical environment, it was also introduced to the free form area. A free form feature is a feature embedded in a single or a set of free form surfaces[5]. The boundary of the features consists of curve segments that may lie within a surface. Poldermann and Horváth[6] provided a general classification of free form surface features based on four major classes: Primary surface features, Modifying surface features, Transition surface features and Auxiliary surface feature. Recently, another free form feature taxonomy was brought forward by De Martino *et al.*[7] and Fontana *et al.*[4]. In their works, according to the shape and different contribution to the free form surface, the free form features were divided into two main categories: shape deformation and shape elimination feature.

In the reverse engineering area, to identify free form features from an existing surface is a key issue. Examples of feature identification from Constructive Solid Geometry (CSG) models were conducted by Lee and Fu[8] and Perng *et al.*[9]. For Boundary-Representation (B-Rep) models, two major approaches are used: Syntactic pattern recognition and the graphics based method[10]. Recently, free form features were applied in reverse engineering since:

1. Much better results can be obtained if features are treated separately than if global surfaces are fitted across the whole data set [11,12],
2. High level entities and parameters can be directly specified and changed by designer[2],
3. Reusing of free form features form existing physical or CAD model will greatly reduce the design work.

For getting the high-level parameters of a free form feature, matching existing data with a pre-defined template is a useful method. With a parameterized template, free form surface information can be directly transferred into high-level parameters for later reusing or editing. Thus, Li and Hui[8] applied free form templates in feature recognition. Recently, some Computer-Aided Industrial Design (CAID) systems emerged[13,14], each of which was in some way based on surface features. Some systems were dedicated to specific types of features, such as protrusions and depressions[15,16]. Surazhsky and Elber[17] developed a metamorphosis process, which is defined as gradual and continuous transformation of one key shape into another. To free form digitized surface, in Spanjaard's work[18], a surface is digitized and a ridge template was used to fit the surface. In the authors' former works[2,19,20,21,22], several free form features had been defined and parameterized.

With free form features, reconstruction of a free form surface from physical model can be carried as follows: The model surface is digitized with a 3-Dimensional (3-D) scanner or Coordinate Measuring Machine (CMM). A point cloud is created which contains the model surface information. Then the Region Of Interest (ROI) is selected by the designer. With a free form feature template, the selected point set is fitted with a specified measuring method, and the parameters of this feature are got. Then, the feature is bridged with the original surface. With the above steps, the free form surface is reconstructed. If the designer wants to change parameters of a feature, only this feature needs to be reconstructed.

Complicated free form surfaces may contain a lot of free form features and parts of them are interfered by other features. Identifying each interfered feature and extracting high-level parameters is an interesting topic for constructing the surface again by Boolean-like operations. The key problem of identifying interfered free form features is that no explicit boundary of free form features can be found. Most of the former works focused on interference of mechanical features, few paper mention interference of free form features. Ganesan and Devarajan[23] proposed a new approach to solve the problem of intersecting features extraction. In Perng and Chang's work[24], a method to solve the feature interaction problems encountered in part-editing is proposed. Generally, although identifying interference and reconstructing interfered free form features is a crucial issue in reconstructing complicate free form surface, it is still an unsolved problem.

In this paper, matching complicated free form surfaces in 3-D space is studied. Several simple free form feature templates are proposed first, where a Hausdoff-like distance is brought forward as a key element in the shape matching function. To isolated free form features, the matching process can be easily carried out by the above optimization function. When there is interference between features, with numerical experiments, an interference threshold is defined. If the interference is lower than this threshold, a free form feature can be identified. The matching algorithms were tested and parts of free form features can be identified with specified conditions.

2. Free form features and parameterization

According to De Martino *et al.*'s work[6], free form features were mainly categorized to structural features and detail features in aesthetic design. Detail features can be divided into two categories: shape deformation and shape elimination feature with different deformation laws. The feature types include step-up, step-down, cavity, bump, n-groove, n-rib, inlet, hole and outlet. In this paper, only detail features are studied. Several simple free form features in a free form surface are shown in **Figure 1**. In the figure, there are three features: a bump, a ridge and a hole where the bump is an isolated feature whereas the ridge intersects the hole.

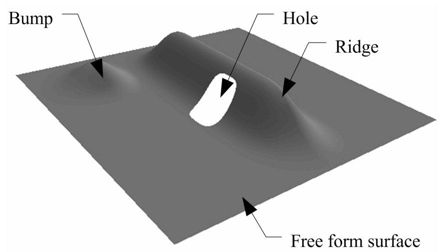


Figure 1: Free form features and interference of features

A feature type (or feature, or feature class) t is specified by a mapping:

$$G_t : Q_t \to 2^{\mathbb{R}^3} \quad (1)$$

where $\mathbb{R}^3$ is the ambient space in the application, $2^{\mathbb{R}^3}$ is the power set (i.e. the set of all subsets) of $\mathbb{R}^3$, the set $Q_t = C_1 \times C_2 \cdots C_m$, called the parameter domain of G_t. Typically C_i represents the domain of a continuous scalar variable q_i (such as a dimension or an angle), but in general C_i can be any set. For a given $q \in Q_t$, $G_t(q)$ specifies a subset in $\mathbb{R}^3$ referred to as feature instance, or pattern, of type t.

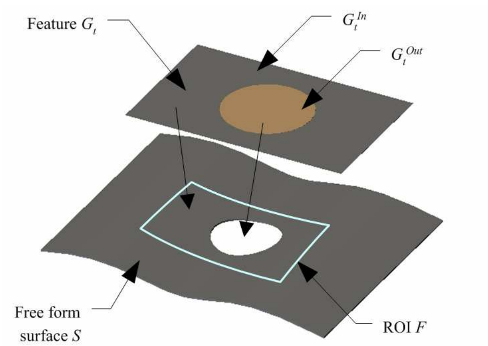


Figure 2: Surface, ROI and Feature instance

Given shape $S \subset \mathbb{R}^3$ as **Figure 2**, the ROI is F, where $F \subset \mathbb{R}^3$. With a specified feature t, a pattern is created as the feature instance in **Figure 2**, by **Equation (1)**, the feature $G_t(q)$ may contain two parts as:

$$G_t(q) = G_t^{In}(q) \cup G_t^{Out}(q) \quad (2)$$

where both $G_t^{In}(q)$ and $G_t^{Out}(q)$ belong to the feature. In **Equation (2)**, the $G_t^{In}(q)$ part represents some characteristic of the feature that is supposed to be located on the shape *S*. For $G_t^{Out}(q)$, this part represents some characteristic of the feature that is supposed to be exterior to *S*. For shape deformation features, $G_t^{Out}(q)$ always equals to $\varnothing$ while in shape elimination features, normally, $G_t^{Out}(q)$ is used to make up for the "lost" data of the surface. For example, for a hole feature as in **Figure 2**, the boundary $G_t^{In}(q)$ ought to fit F whereas $G_t^{Out}(q)$ ought to be located in the hollow space of the shape surface.

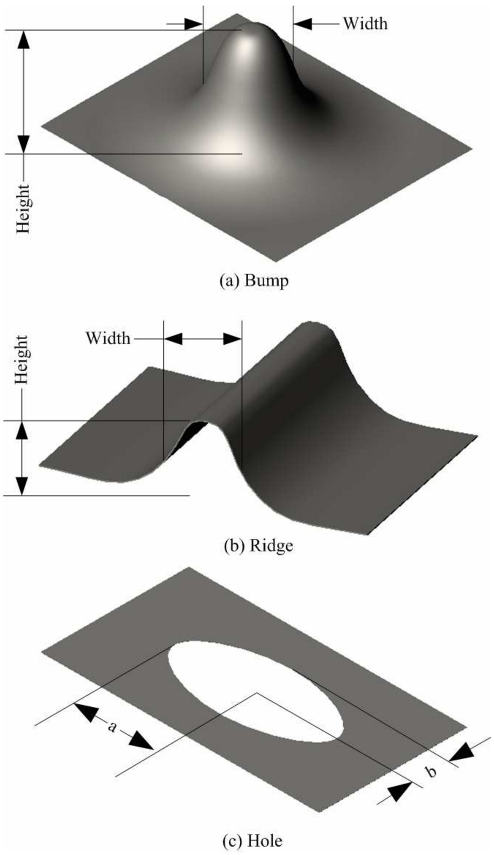


Figure 3: Free form feature templates

When a feature is applied in fitting, it should be parameterized and a mathematical representation should be generated. The complexity of a feature depends on the application at hand. Of course, the more parameters are used, the more flexible a feature is, but problems appears as follows[2]: first, too many parameters will confuse the designer who prefers to deal with five to eight parameters at a time. Second, more parameters will make the optimizing procedure much more difficult or even impossible. In the proposed research, normally, two to four parameters are used to represent the shape of a free form feature and additional six parameters are used to represent the position and orientation of the feature. With more flexible definitions of the feature template, those free form features are simplified and categorized to Bump, Ridge and Hole as shown in **Figure 3**. With these definitions[26], many free form features can be derived, for example, if the height of the ridge is negative, it turns into a groove.

With the high-level definitions as **Figure 3**, refer to the author's former works[22] and Non-Uniform Rational B-Spline (NURBS) representations[27], a feature template can be represented as:

$$G_t^{In}(q) = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} N_{i,p}(u)N_{j,r}(v)\omega_{i,j}(q)P_{i,j}(q)}{\sum_{i=0}^{m}\sum_{j=0}^{n} N_{i,p}(u)N_{j,r}(v)\omega_{i,j}(q)}, \text{ and}$$

$$G_t^{Out}(q) = \begin{cases} \dfrac{\sum_{k=0}^{e}\sum_{l=0}^{f} N_{k,p}(u)N_{l,r}(v)\omega_{k,l}(q)P_{k,l}(q)}{\sum_{k=0}^{e}\sum_{l=0}^{f} N_{k,p}(u)N_{k,r}(v)\omega_{k,l}(q)} & \text{Bump Ridge}, \\ \varnothing & \text{Hole} \end{cases} \quad (3)$$

where in the NURBS representations, the control points $P(q)$ and weights $\omega(q)$ are the functions of the parameters of free form features. The representation of feature template can be a surface model or a point sample obtained by the iso-parameter method[26].

3. Template matching

With the feature templates, for matching the existing free form features in 3-D space, a similarity measurement method ought to be selected to measure the difference between the original shape and the free form feature template.

With a specified feature, according to **Equation (2)**, the matching procedure is that finding $q \in Q_t$ for

$$\min_{q\in Q_t}(d(G_t^{In}(q),F) - \lambda d(G_t^{Out}(q),F)),\ 0 < \lambda \le 1, \quad (4)$$

where d is a difference criterion (or similarity measure) for two subsets of $\mathbb{R}^3$. In the equation, $d(G_t^{In}(q),F)$ is the similarity measurement between part $G_t^{In}(q)$ of the feature and F. According to the definitions, theoretically, it may be zero when the feature exactly fits the surface. To $d(G_t^{Out}(q),F)$, it is the difference between $G_t^{Out}(q)$ and F. It is easily to find that after the feature exactly fits the surface, there is a local maximum of $d(G_t^{Out}(q),F)$. By the scalar coefficient λ, the "weight" of $d(G_t^{Out}(q),F)$ can be adjusted in the similarity measurement. The whole matching procedure can be accelerated with different λ[26]. When **Equation (4)** is optimized, it shows whether the feature had exactly fit the given surface.

Equation (4) delivers the feature instance of type t that matches F optimally. The following assumptions are made in this paper:

1. Shape S approximates (a part of) the boundary of a 3-D object; typically S is a discrete point set originating from 3-D surface scanning.
2. Part of the feature instance $G_t^{In}(q)$ represents (a part of) the boundary surfaces of a 3-D solid. It can be represented as a collection of surfaces. However, degenerate cases such as the collapsing of the feature instance into a curve or a point are not excluded.
3. On the contrary, another part of the feature instance $G_t^{out}(q)$ represents part of the feature but not on the boundary of a 3-D solid. It is generated by the "natural" definition of the feature. Such as the hole feature in **Figure 2**, the center of the hole data is generated by interpolation of the hole boundary.

The problem can be extended to search among multiple feature types t (the learning set) to find the best fit to shape F. An appropriate similarity measure d for **Equation (4)** should be specified as a tool for the fitting process. There were many similarity measurements defined in the literature[28,29]. It can be observed that a distance can be defined as

$$d(A,B) = \max_{a\in A} \max_{b\in B} |a-b| \; . \tag{5}$$

As describe by the author's former work[22,26] the Directed Hausdorff Distance (DHD) has been introduced to cure some drawbacks in **Equation (5)**. The Hausdorff Distance $D(A,B)$ between the shapes A and B is

$$D(A,B) = \max(H(A,B), H(B,A)) \; , \tag{6}$$

where $H(A,B) = \sup_{r\in A}(\inf_{s\in B} |r-s|)$, here $|r-s|$ denotes the Euclidean distance between the points s and r. To reduce the sensitive to noise and inaccuracies in the shape data[20], the Mean Directed Hausdorff Distance (MDHD) $M(A,B)$ is introduced as:

$$M(A,B) = \iint_A \inf_{s\in B} |r-s| \, dA \Big/ \iint_A dA \; , \tag{7}$$

where the integration is over the surface of A, normalized by the surface area of A. Applying MDHD, **Equation (4)** changes to

$$\min_{q\in Q_t}(M(G^{In}(q),F) - \lambda M(G^{Out}(q),F)) \; . \tag{8}$$

As the surface data may come from a 3-D scanner or a CMM, it can be treated as a point set. In this paper, point sets are selected as the basis to the dissimilarity computation. If both the original surface F data and the features $G_t(q)$ are digitalized, points P_i^{In}, P_i^{Out} and P_i^F in $G_t^{In}(q)$, $G_t^{Out}(q)$ and F respectively, are available:

$$\{P_i^{In} \in G_t^{In}(q) \mid i = 1,m\} \; ,$$
$$\{P_i^{Out} \in G_t^{Out}(q) \mid i = 1,n\} \text{ and}$$
$$\{P_i^F \in F \mid i = 1,k\} \; . \tag{9}$$

The MDHD in **Equations (8)** are then approximated by

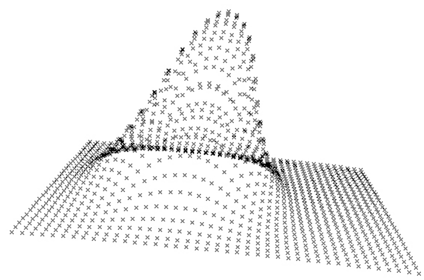
$$M(G_t^{In}(q),F) = \frac{1}{m} \sum_{i=1,m} (\min_{j=1,k} |P_i^{In} - P_j^F|) \; ,$$

$$M(G_t^{Out}(q),F) = \frac{1}{n} \sum_{i=1,n} (\min_{j=1,k} |P_i^{Out} - P_j^F|) \; . \tag{10}$$

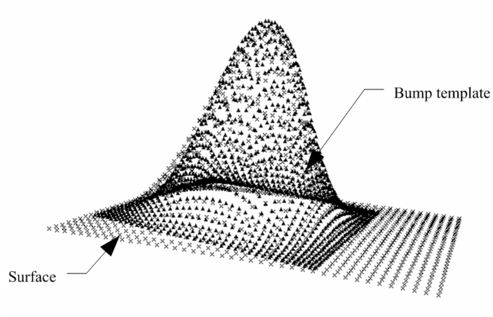
A matching procedure is required to obtain the proper parameters of the feature template $G_t(q)$ under variation of the parameters $p_{Opt} \in C$, where C is the fitting parameters set. Then, the search for the optimized parameters $p_{opt} \in C$ is defined as:

$$p_{opt} = \operatorname{Arg}\min_{p\in C}(M(G_t^{In}(q),F) - \lambda M(G_t^{Out}(q),F)) \; . \tag{11}$$

Equation (11) is named optimization function, which is applied as the objective function in a fitting procedure. With **Equation (11)** as the optimization function, by ACIS® and Open Inventor®, the whole system was modeled by Visual C++® and the search procedure was conducted by means of the IMSL® C numerical libraries. In **Figure 4(a)**, a ROI of a free form surface which contains a bump feature is shown. With the defined bump template and optimization function, a matching procedure was carried out and **Figure 4(b)** shows the final fitting result.



(a) ROI of a free form surface containing a bump feature

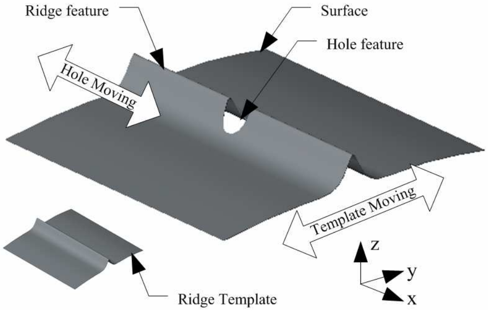


(b) Fitted bump feature

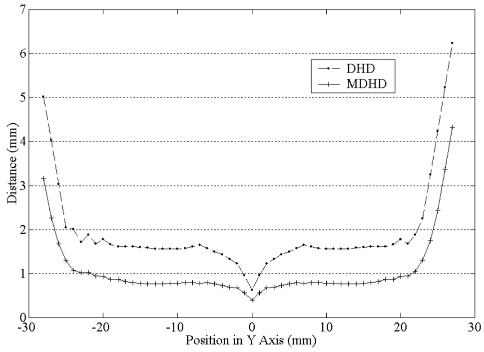
Figure 4: Matching a bump feature with a feature template

4. Feature interference

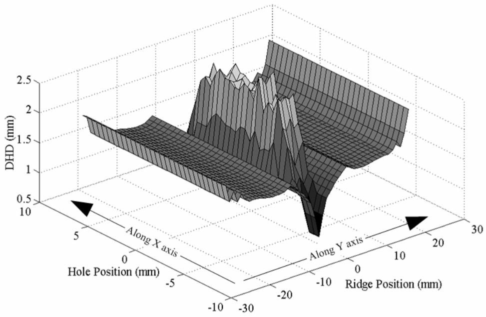
Feature interference means that one feature partially overlaps another feature. A complicated free form surface may be constructed by many free form features. Identifying feature interference is an important topic in reusing existing designs. For example, in **Figure 1**, a ridge is interfered by a hole. If the designer wants to reuse the shape of the ridge feature, the influence of the hole should be removed.



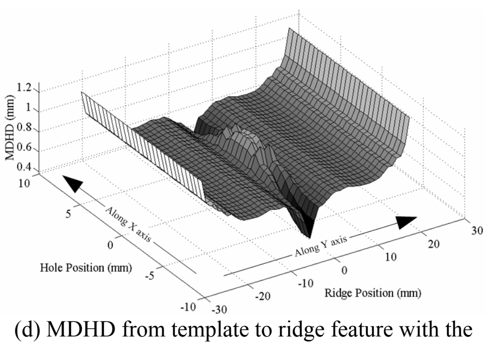
(a) Set up of experiment



(b) DHD and MDHD from template to a ridge feature



(c) DHD from template to ridge feature with the interference of a hole feature



(d) MDHD from template to ridge feature with the interference of a hole feature

Figure 5: Interference of a ridge and a hole

When feature interference occurs, shape information of this feature is partly lost or changed by the other features. Suppose a feature is interfered by another. If the other feature is a hole, it can be treated as "lost" data; if the other feature is a deformation feature, it can be treated as noise. Here, an interference threshold is defined as:

$$\psi = \sum_{i=1}^{n} \frac{f(T(G_t^i(q)))}{f(G_t^{main}(q))}, \tag{12}$$

where $G_t^{main}(q)$ is the feature being interfered in ROI, $G_t^i(q)$ are a series of features interfering $G_t^{main}(q)$, $i = 1,2,\cdots n$ represents the number of features which interfere $G_t^{main}(q)$. Generally, f is an abstract function based on the dimension of the interfered part of the freeform features, such as diameter, surface area or volume. T is a trimming function, which will trim $G_t^i(q)$ by ROI. The value of ψ reflects the "degree" of interference, that is, for given features and function f, the large ψ, the greater the interference is. To a fixed condition, if ψ is larger than a particular threshold value, the interfered free form feature can not be identified.

Suppose a surface area is selected as f, with NURBS representation as follows,

$G_t^i(q) = s_t^i(u,v)$ $0 < u < 1$, $0 < v < 1$, and

$$G_t^{main}(q) = s_t^{main}(u,v),\ 0 < u < 1,\ 0 < v < 1, \tag{13}$$

where u,v are the parameters of NURBS representation. With trim function T,

$$T(G_t^i(q)) = s_t^i(u,v),\ u_1^i < u < u_2^i,\ v_1^i < v < v_2^i, \tag{14}$$

where $0 < u_1^i < u_2^i < 1$, $0 < v_1^i < v_2^i < 1$. Then **Equation (12)** changes to

$$\psi = \sum_{i=1}^{n} \frac{\int_{u_1^i}^{u_2^i}\int_{v_1^i}^{v_2^i} d(s_t^i(u,v))dudv}{\int_0^1\int_0^1 d(s_t^{main}(u,v))dudv}. \tag{15}$$

If both the template and the surface are digitized to point sets, they are

$P^i = \{P_j^i \in G_t^i(q) \mid j = 1, m_i\}$ and

$$P^{main} = \{P_j^{main} \in G_t^{main}(q) \mid j = 1, m_{main}\}. \tag{16}$$

Here, diameter, which is the largest Euclidean distance between any two points in a point set, is selected as function f, thus **Equation (12)** changes to

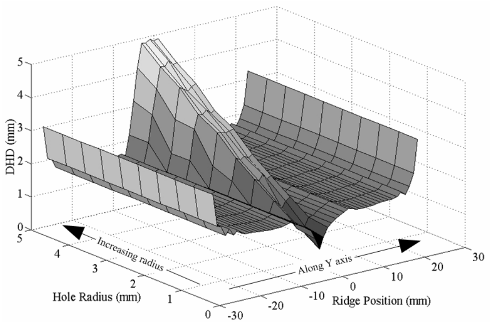
$$\psi = \sum_{i=1}^{n} \frac{Diam(T(P^i))}{Diam(P^{main})}, \tag{17}$$

where $Diam$ is the diameter function.

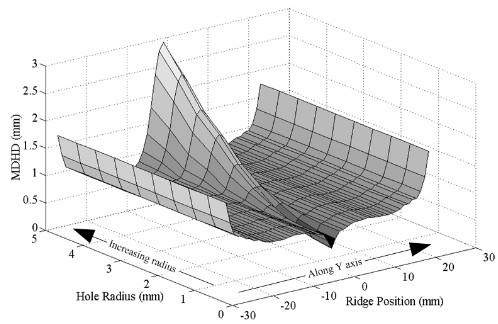
Figure 5 shows the set up of an interference experiment between a ridge feature and a hole feature. The length of the ridge template is 10 mm where the height and width exactly match the height and width of the ridge feature on the surface with the optimizing method mentioned in the above section. Since the ridge is a shape deformation feature, the optimization function is simplified to DHD and MDHD.

Without the interference from the hole feature, when the ridge is moved along the Y axis (see **Figure 5(a))**, the DHD and MDHD from surface to ridge template is shown in **Figure 5(b)**. In the figure, obviously, there is a minimum in the center which means the matching process can be successfully carried out with the optimization function.

Due to a hole feature with 2.5 mm radius in the surface which is right cross the ridge feature, feature interference occurs. Moving the hole along the X axis while moving the ridge template along Y axis in produces the DHD and MDHD from the surface to the template as shown in **Figure 5(c)** and **Figure 5(d)**. Since the length of the ridge template is 10 mm, and when the hole is positioned between –5.0 to 5.0 mm, the hole "cuts" into the ridge. In the figures, the minimum of both DHD and MDHD disappear when the hole fully "cuts" in the region where would be the optimum position of the template. The "no minimum" region of MDHD is a little bit smaller than DHD since MDHD is better than DHD, suppressing the noise and loss of data[20].



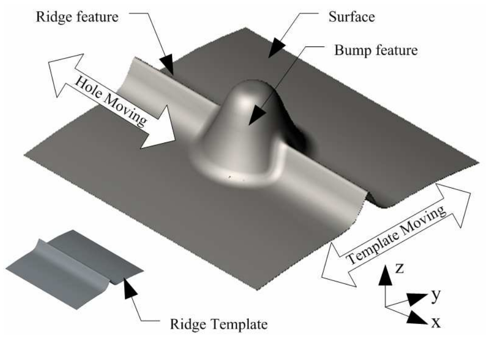
(a) DHD from the template to an interfered ridge



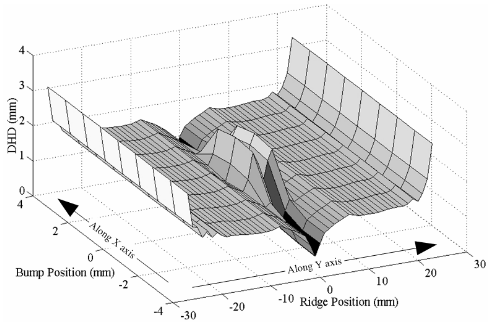
(b) MDHD from the template to an interfered ridge

Figure 6: Interference of a ridge with a chaining radius hole

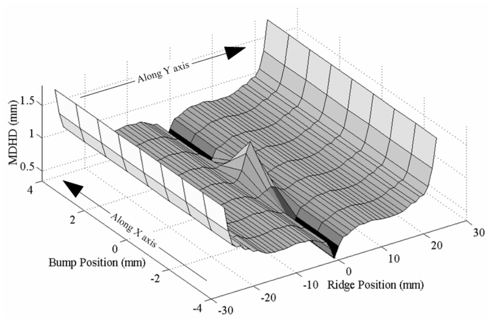
To find the possible radii of the hole which have no influence on the optimization procedure, the hole is located in the center of the ridge at X = 0 mm, Y = 0 mm in **Figure 5(c)**. As a function of the radius of the hole (from 0 to 5mm) and varying the ridge position along Y axis, the results of DHD and MDHD are shown in **Figure 6(a)** and **Figure 6(b)** respectively. For the result of DHD, the minimum disappears when the radius reaches 1.0 mm where to MDHD, it happens at 1.2 mm. The results also verify the conclusion that MDHD can effectively suppress the noise and loss of data. Let the diameter be the abstract function f , with **Equation (17)**, it is easy to get that ψ =0.152 for DHD and ψ = 0.182 for MDHD are the threshold value. That is, if the interference of a ridge and hole feature is less than 0.182, the feature can be identified and the optimizing procedure can be undertaken.



(a) Set op of the experiments



(b) DHD from the template to an interfered ridge

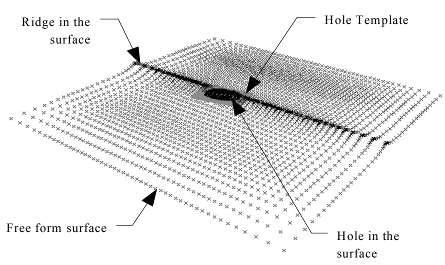


(c) MDHD from the template to an interfered ridge

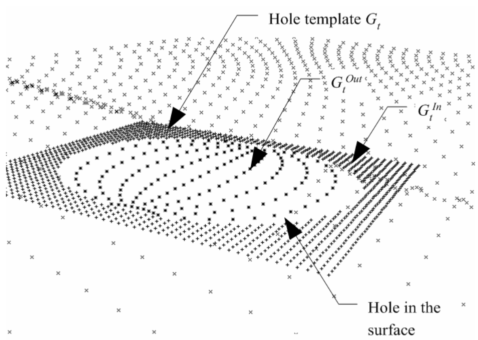
Figure 7: A ridge feature interfered by a bump feature

A similar experiment was set up to demonstrate the interference of a bump feature and a ridge feature. **Figure 7** shows the set up and results of the experiment. A bump with a width of 40 mm and height of 40 mm is set to cross the ridge feature. Both the bump and the ridge motion follow the direction individually shown in **Figure 7(a)**. The behaviors of DHD and MDHD are shown in **Figure 7(b)** and **Figure 7(c)** respectively. To the interfered ridge feature, different from interfered by a hole, the interference from a bump can be treated as "noise" since the bump is a shape deformation feature. From the results, it shows that if the size of the bump exceeds a particular value, both the minimum of DHD and

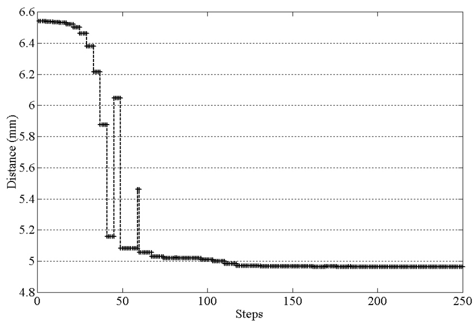
MDHD disappear when the bump is fully moved to the center of the ROI. Like for the hole, the "no minimum" area of MDHD is much smaller than DHD.



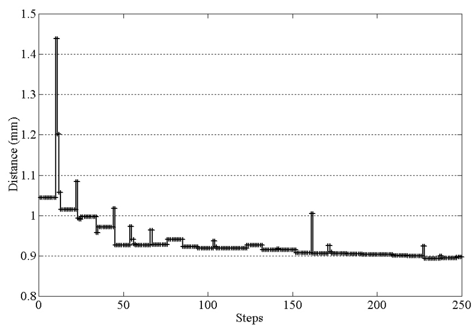
(a) Matching a interfered hole



(b) Zoomed matching result



(c) Optimizing procedure of matching Step-2



(d) Optimizing procedure of matching Step-5

Figure 8: Matching a interfered hole feature

5. Matching an interfered feature

With feature interference analysis in the above section, it shows that interfered free form feature template matching can be carried out when the interference is less than ψ. In this section, a numerical matching experiment demonstrates the possibility of matching free form features when it is interfered with the other features.

The numerical experiment is set to fit a hole that is interfered by a ridge feature as in **Figure 8(a)**. First the free form surface was digitalized to 2410 points. The hole template was digitalized to 1369 points. In the optimizing procedure, the quasi-Newton method, the coefficient λ and the searching strategy have been discussed in the authors' former works[22,26]. At beginning of the search procedure, the template was put on an arbitrary position in 3-D space, then the following strategy was applied:

Step-1: Fit the translation of the template. λ in the optimization function is set to 0.1 in order to emphasize $G_t^{Out}(q)$,

Step-2: Fit the orientation of the template. λ in the optimization function is also set to 0.1 in order to emphasize $G_t^{Out}(q)$,

Step-3: Fit the translation of the template again. λ in the optimization function is set to 0.6 in order to emphasize both $G_t^{Out}(q)$ and $G_t^{in}(q)$,

Step-4: Fit the long axis ***a*** and short axis ***b*** of the template. λ in the optimization function is also set to 0.6,

Step-5: Fit the all the parameters simultaneously. λ in the optimization function is also set to 0.6.

To prevent extreme long fitting times each stage in a fit has a timeout value. This timeout forces the process to go to the next stage in case the minimizer cannot terminate within a particular amount of time. **Figure 8(a)** shows the finial result of the fitting result where a zoomed figure of the fitting area is shown in **Figure 8(b)**. With a Pentium III 700MHZ computer, the optimizing procedure was completed around 1250 steps in 10 minutes. In the results, the long axis radius ***a*** of the hole is 3.22mm where the long axis radius ***a*** is 2.56 mm. **Figure 8(c)** and **Figure 8(d)** show the optimizing procedure of Step-2 and Step–5. From the figures, it is very clear that the optimization converges well, although the hole is interfered by the ridge.

6. Conclusion and future works

In this paper, definitions and implementations of two shape dissimilarity measures in full 3D are introduced with the emphasis on matching a feature which has interference with others. From results of numerical experiments, the optimization function using MDHD was the preferred measure for interfered feature matching. Analysis of the shape distance as functions of the parameter components revealed that robust shape matching was feasible with some pre-defined condition.

Current research is directed towards more complicate conditions of matching free form features. The extension to different types of feature templates, such as a hybrid free form feature template matching, different effects of shape deformation and shape elimination features on feature interference are under investigation. In the conceived hybrid template, the complexity of template is increased step by step. In each step, a new free form feature template will be added to this hybrid template, the shape similarity of the template and the free form surface to be matched is also increased step by step. In the matching process, the major feature would be matched with the template first, then the second major feature is added, then the third. The parameter ψ in **Equation (12)** can be used to decide which feature is the major one. Generally, for a given free form feature definition and function f, the feature with the smallest ψ is the major feature.

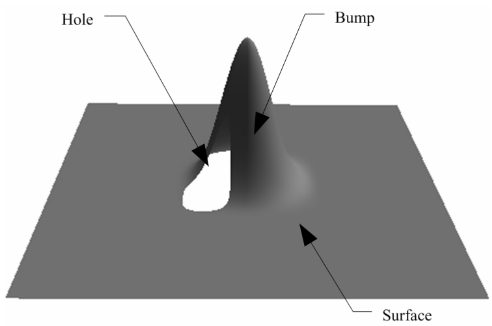


Figure 9: A hybrid free form feature template

Figure 9 shows a hybrid feature template. In the figure, a bump template is applied first in the matching. Then a hole feature will be added to the hybrid template.

Acknowledgments

The research described in this paper is supported by ICA project of Faculty of Design, Engineering and Production, Delft University of Technology. Assistance from ICA team members is appreciated.

Reference

1. Ingle K. A., *Reverse Engineering*, McGraw-Hill, New York, 1994.
2. Vergeest J. S. M. and Horváth I., Parameterization of freeform features, *Proceeding of Shape Modeling International*, IMA-CNR, IEEE, Piscataway, 20-29, 2001.
3. Shah J. J. and Mantyla M., *Parametric and featured based CAD/CAM*, Wiley-Interscience Publication, John Wiley Sons In., 1995.
4. Fontana M., Giannini F. and Meirana M., A free form feature taxonomy, *Proceeding of Eurographics '99*, 18 (3), 1999.
5. Li, C. L. and Hui, K. C., Feature recognition by template matching, *Computers & Graphics*, 24, 569-582, 2000.
6. Poldermann and Horváth, surface based design based on parametrized surface features, *Proceedings International Symposium on Tools and Methods for Concurrent Engineering*, Institution of Machine Design, Budapest, 432-446, 1996.
7. De Martino T., Fontana M., Giannini F. and Meirana M., *A free form feature taxonomy*, Rapporto Tecnico IMA (8), 1998.
8. Lee Y. C., Fu K. S., Machine understanding of CSG: extraction and unification of machining features. *IEEE Computer Graphics and Applications*, 7 (1), 20-32, 1987.
9. Perng D. B., Chen C. and Li R. K., Automatic 3D machining feature extraction from 3D CSG solid input. *Computer-Aided Design*, 22 (5), 285-295, 1990.
10. Stefano P., Automatic extraction of form features for casting. *Computer-Aided Design*, 29 (11), 761-770, 1997.
11. Varady T., Martin R. R. and Cox J., Reverse engineering of geometric models--an introduction, *Computer-Aided Design*, 29 (4), 255-268, April 1997.
12. Varady T. and Martin R.R., Reverse engineering, *Handbook of CAD*.
13. Rowe J., Surface modeling. *Computer Graphics Wold*, 20 (9), 47-52, 1997.
14. Wyvill G., McRobie D. and Gigante M., Modeling with features. *IEEE Computer Graphics and Applications*, 15 (5), 40-46, 1997.
15. Cavendish J. C., Integrating feature based surface design with free form deformation, *Computer-Aided Design*, 27 (9), 703-711, 1995.
16. Van Elsas P. A. and Vergeest J. S. M., Displacement feature modelling for conceptual design, *Computer-Aided Design*, 30 (1), 19-27, 1998.
17. Surazhsky T. and Elber G., Matching free-form surfaces, *Computers & Graphics*, 25 (1), 3-12, February 2001.
18. Spanjaard, S., *Documentation of ridge fit application and its source code. Technical report*, Faculty of Design, Engineering and Production, Delft University of Technology, Delft, 2000.
19. Vergeest J. S. M., Spanjaard S., Horváth I. and Jelier J. J. O., Fitting Freeform Shape Patterns to Scanned 3D Objects", *Journal of Computing and Information Science in Engineering*, Transactions of the ASME, 1 (3), 218-224, 2001.

20. Vergeest J. S. M. and Horváth I., Matching 3D freeform shapes to digitized objects and its applications in shape modeling. In: M. Hamdy Elwany (Ed.), *Proc. PEDAC2001 7th Int. Conf. on Product Engineering, Design and Control*, Production Engineering Department, Alexandria University, Alexandria, 2001.

21. Vergeest J. S. M. and Spanjaard S., The computation of 3D pattern matching as a tool for conceptual freeform shape design. *Proc. of the 2001 Design Automation Conference*, DETC'01/DAC-21139, ASME, New York, 2001.

22. Vergeest J. S. M., Spanjaard S. and Song Y., Direct 3D pattern matching in the domain of freeform shapes. V. Skala (Ed.), *Proc. of the WSCG'2002*.

23. Ganesan R. and Devarajan V., Intersecting features extraction from 2D orthographic projections, *Computer-Aided Design*, 30 (11), 863-873, 15, 1998.

24. Perng D. B. and Chang C. F., Resolving feature interactions in 3D part editing, *Computer-Aided Design*, 29 (10), 687-699, 1997.

25. Braquelaire J. P. and Domenger, J. P. Geometrical, topological, and hierarchical structuring of overlapping 2-D discrete objects, *Computers & Graphic*s, 21 (5), 587-597, 1997.

26. Song Y., Vergeest J. S. M. and I. Horváth, Reconstruction Free Form Surface With Parameterized features, *Proc. of the 2002 Design Automation Conference*, DETC'02/DAC, ASME, New York, 2002. (*In press*)

27. Piegl L. and Tiller W., *The NURBS book*, Springer pres, Berlin, 1996.

28. Veltkamp, R. C., Shape matching: similarity measures and algorithms, *Proc. Shape Modeling International Conference*, a. Pasko and M. Spagnuolo, Eds., Los Alamitos, IEEE, 188-197, 2001.