



Poisson based reuse of freeform features with NURBS representation

Wei Zhao, Shuming Gao^{*}, Yusheng Liu, Hongwei Lin

State Key Lab. of CAD&CG at Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Article history:

Received 10 February 2007
 Received in revised form 27 June 2008
 Accepted 4 September 2008
 Available online 7 November 2008

Keywords:

Feature reuse
 Freeform feature
 NURBS
 Poisson equation
 CAD

ABSTRACT

The effective reuse of freeform features represented by NURBS is still an open issue. In this paper, a novel approach to the reuse of freeform features with NURBS representation is proposed. Firstly, the conditions for preserving differential properties of reused freeform features are derived, and a reuse-oriented representation of freeform features is put forward. Based on them, a reuse algorithm of freeform features is developed, which adopts Poisson equation, knots insertion and degree elevation to achieve the preservation of the differential geometry properties and the adaptability. The approach is implemented and some examples are given.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

As we all know, product design can be classified into three categories: variant design, adaptive design and innovative design. It is estimated that about 80% designs are variant design and adaptive design. The most important factor making variant design and adaptive design effective and efficient is the design reuse. Design reuse, in brief, is to use the existent design data and design knowledge as much as possible to make new designs. Obviously it can remarkably shorten the design cycle and improve the design quality.

Product design is one of the most complicated design activities. As market competition becomes more and more fierce, the cycle of product design has to be increasingly shortened, which forces companies and designers to fasten product design through design reuse. Therefore design reuse has been paid more and more attention by both industries and academics.

Among various design reuses, design feature reuse is one of the most important design reuse for product design. This is because, the design feature, referring to the form with certain function significance, is the basic design unit which the designers are used to utilizing, and thus is the ideal reuse object. In general, feature can be divided into two classes: regular features and freeform features. The former are those features whose boundary surfaces

are planes or quadric surfaces and the latter are those features whose boundary surfaces are freeform surfaces. As for the reuse of regular features, design by feature approach [1] has already provided a solution mechanism to some extent. However, due to the high complexity of freeform features, freeform feature reuse is still a challenging issue. To realize the reuse of freeform features, *i.e.* copy a freeform feature on a surface (source surface) and paste it onto another surface (target surface), a few works [2–6] have been conducted in recent years. These works focus on the reuse of the freeform features represented by mesh, implicit surface or point cloud, without considering the reuse of the freeform features with NURBS representation up to now although NURBS has been used as the fundamental representation of freeform shapes in all commercial CAD systems. Obviously it is more difficult to realize the high-quality reuse of the freeform features with NURBS representation, in which the differential geometry properties of the reused freeform feature on target surface are kept the same as those of the original freeform feature on source surface (called preservation of differential geometry properties hereafter) and the reused freeform feature adapts to the geometry of the target surface (called adaptability hereafter). This is because the shape of NURBS surface is controlled by many factors including control points, knot vectors, degrees and weights, and all these factors must be taken into account in order to achieve the high-quality reuse of the freeform features.

The objective of this work is to propose a novel approach to reuse of the freeform features with NURBS representation, which can guarantee the high quality of the feature reuse, *i.e.* achieve the preservation of differential geometry properties and adaptability.

^{*} Corresponding author. Tel.: +86 571 88206681x514; fax: +86 571 88206680.
 E-mail addresses: zhaowei@cad.zju.edu.cn (W. Zhao), smgao@cad.zju.edu.cn (S. Gao), ysliu@cad.zju.edu.cn (Y. Liu), hwlin@cad.zju.edu.cn (H. Lin).

And with the approach, designers can effectively reuse the freeform features represented by NURBS from one surface to another.

2. Related works

The research on the reuse of NURBS based freeform features can be traced back to Barghiel et al. [7]. In their study, a local B-Spline surface is hierarchically pasted onto another B-Spline surface by setting up the mapping between the control points and their Greville abscissa and the mapping between the two-parameter domain of the two surfaces. This method suffers from the bad continuity between the source surface and the target surface and further refined by Conrad and Mann [8] to guarantee C^1 continuity using quasi-interpolation to determine the boundary control points of the feature. In this kind of method, the whole source surface is regarded as a single geometric feature and there is no effective transition between the feature and the target surface.

In order to support their freeform feature based design system, van der Berg [9] proposed an attachment operation which attaches a face of a new volumetric freeform feature to a face of a model. In their work, the feature is firstly parameterized by configurations of freeform definition points (FFDPs). By using geometric constraints on these FFDPs and other geometric entities within the model, freeform attachments can be realized. A limitation of attachment operation is that only volumetric wrap features generated by sweeping and skinning operations can be attached.

Zhao et al. [10] proposed a freeform feature reused method based on Poisson equation. This method first constructs similar distribution of control points on target surface with that of the freeform feature on source surface. By solving Poisson equation, related control points on target surface are repositioned and the reuse is achieved. The representation of the freeform feature, odd uniform B-Spline, limits its usage.

3. Conditions for preserving differential geometry properties of reused freeform features

The differential geometry properties of a surface reflect the characteristics of its shape, e.g. the continuity and the smoothness. So preservation of the differential geometry properties during freeform feature reusing is very critical for obtaining a high-quality reuse result. As we know, the differential geometry properties of a NURBS (e.g., tangent, curvature and torsion, etc.) are all expressed by its derivatives, e.g. for a NURBS curve, its tangent is the first derivative, thus to preserve the differential geometry properties of a NURBS is to preserve its derivatives.

Based on above analysis, we derive the conditions for preserving differential properties of reused features as follows (the detailed derivation is given in the Appendix A):

Conditions for preserving differential properties of reused freeform features: given the source surface $S(u, v) = \sum \sum N_{i,p}(u)N_{j,q}(v)P_{i,j}$ with knot vector $\{u_i, v_i\}$ and weights $\{w_{i,j}\}$, and the target surface $S'(u, v) = \sum \sum N_{i,m}(u)N_{j,n}(v)P'_{i,j}$ with knot vector $\{u'_i, v'_j\}$ and weights $\{w'_{i,j}\}$, the conditions for preserving differential properties between the source surface and target surface are as follows:

- (1) their degrees in both u and v directions are equal, i.e. $p = m$ and $q = n$;
- (2) their forward differences of weighted control points are equal, i.e. $\nabla \omega_{i,j} P_{i,j} = \nabla \omega'_{i,j} P'_{i,j}$;
- (3) their forward differences of weights are equal, i.e. $\nabla \omega_{i,j} = \nabla \omega'_{i,j}$;
- (4) the forward difference of knots in both directions are equal, i.e. $\nabla u_i = \nabla u'_i$ and $\nabla v_i = \nabla v'_i$.

In above conditions, the first and fourth can be made satisfied by performing degree elevation and knot insertion operation on target surface and/or source surface if they are not met at the beginning. And since the weights of a NURBS surface can be modified directly, the third condition can be easily got met by assigning the weights of every control point on source surface to the corresponding control point on target surface. However, the second condition is hard to meet in most cases because the forward differences of weighted control points of the source surface and target surface are always different, e.g. one is a plane and another is a curved surface or their orientations are different. In this paper, this problem is solved using an optimization method whose optimization objective is defined as

$$\min \|\nabla \omega_{i,j} P_{i,j} - \nabla \omega'_{i,j} P'_{i,j}\| \quad (1)$$

Note that the forward difference of weighted control points $\nabla \omega_{i,j} P_{i,j}$, in essence, is the gradient of $\omega_{i,j} P_{i,j}$, and the solution of Eq. (1) can be obtained by solving Poisson equation with Dirichlet boundary conditions. For convenience, we still call the weighted control point control point hereafter.

4. Poisson equation

4.1. Formulation

The Poisson equation with Dirichlet boundary conditions is formulated as

$$\Delta f = \text{div}(w) \quad \text{over } \Omega \quad \text{with} \quad f|_{\partial\Omega} = f^*|_{\partial\Omega} \quad (2)$$

where $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2 + \partial^2/\partial z^2$ is the Laplacian operator, and $\text{div}(w) = \partial \omega_x/\partial x + \partial \omega_y/\partial y + \partial \omega_z/\partial z$ is the divergence of guidance vector $\omega = (\omega_x, \omega_y, \omega_z) \in R^3$, Ω is the domain of the unknown function f and f^* is a known function which specifies the value of f on the boundary of Ω , noted by $\partial\Omega$.

It has been proved that the Poisson equation is equivalent to the following least-squares minimization problem:

$$\min_f \int \int_{\Omega} \|\nabla f - \omega\|^2 \quad \text{over } \Omega \quad \text{with} \quad f|_{\partial\Omega} = f^*|_{\partial\Omega} \quad (3)$$

where ∇ is the gradient operator.

4.2. Discrete Poisson solver

In this paper, we set up Poisson equation on control net of target surface using a discrete variational method similar to [11]. Let S be the control net and Ω is the subset of S with boundary $\partial\Omega$. For each control point p in S , let N_p be the set of its 4-connected neighbors and let (p, q) be a control point pair such that $q \in N_p$. Let f_p be the coordinates at control point p . The task is to compute the positions $f|_{\Omega} = \{f_p, p \in \Omega\}$. Fig. 1 illustrates these notations.

The discretization of Eq. (3) yields the following discrete, quadratic optimization problem:

$$\min_f \sum_{p \in \Omega} (f_{pq} - \omega) \quad (4)$$

where f_{pq} is the gradient of p and $f_{pq} = f_p - f_q$. The Eq. (1) implies that the guidance vector field ω is the gradient field taken from the control net of source surface. Suppose g is the control net of source surface, then $\omega = \nabla g$. Therefore, the solution of Eq. (4) satisfies the following linear equation:

$$\sum_{q \in N_p} f_{pq} = \sum_{n \in N_m} g_{mn} \quad \text{over } \Omega \quad f|_{\partial\Omega} = g|_{\partial\Omega} \quad (5)$$

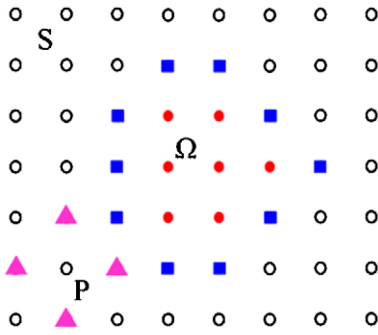


Fig. 1. Notations for discrete Poisson solver. The rectangle control points $\partial\Omega$ are boundary of Ω (solid circles). The triangular points are the 4-connected neighbors of point P , denoted by N_p .

In fact, the both sides of the above equation express the Laplacian of control point which can be formulated as

$$\Delta f_p = \sum_{q \in N_p} f_{pq} = 4f_p - \sum_{q \in N_p} f_q \quad (6)$$

5. Reuse-oriented representation of freeform features

In order to support reuse effectively, a reuse-oriented representation of freeform features is put forward and used in this study. Before describing the specific representation, some properties of NURBS related to this study are introduced first.

5.1. Related properties of NURBS

NURBS is a prevalent mathematical representation for both freeform shapes and standard analytical shapes in CAD application. It has many useful properties and is thus employed by almost commercial CAD systems as a fundamental geometry representation. Among these properties, the following local support properties are closely related to this study.

Property 1. Given knot span $[u_i, u_{i+1})$, only $N_{i,k}(u)$ ($i - k \leq j \leq i$) are nonzero.

The corresponding control point set $\{P_{i-k}, \dots, P_i\}$ of basis function $N_{i,k}(u)$ ($i - k \leq j \leq i$) is called the **complete control point set (CCPS)** of the curve segment $[C(u_i), C(u_{i+1}))$.

Property 2. If modify the control point P_i , only the curve segment $[C(u_i), C(u_{i+k+1}))$ will be changed.

In this study, the interval $[u_i, u_{i+k+1})$ is called the support interval of P_i and P_i is called the **principal control point** of curve segment $[C(u_i), C(u_{i+k+1}))$. Obviously, to reuse a freeform feature, only its principal control points need to be considered. For a given curve segment, the set of all its principal control points is called **principal control point set (PCPS)** of this segment.

Given a curve segment, denoted by $[A, B]$, its CCPS and PCPS can be determined as follows:

- (1) Calculate the parameter interval $[u_a, u_b]$ of $[A, B]$ in the parameter domain of the NURBS curve;
- (2) Find out two knots u_m and u_n from the knot vector which satisfy $u_m \leq u_a \leq u_{m+1}$ and $u_{n-1} \leq u_b \leq u_n$, and determine the CCPS = $\{P_{m-k}, \dots, P_{n-1}\}$ according to the property 1;
- (3) For curve segment $[C(u_m), C(u_n)]$, determined PCPS = $\{P_m, \dots, P_{n-k-1}\}$ according to the property 2.

Taking Fig. 2 as an example, the curve segment $[A, B]$ of the NURBS curve (for convenience, without loss of generality, it is a cubic uniform rational B-Spline curve) is a 2D freeform feature, denoted by FT_{AB} hereafter. We can determine the CCPS and the PCPS of a freeform feature by the method described above. For example, in Fig. 2 the control point sets $\{P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}\}$ and $\{P_5, P_6, P_7\}$ are the CCPS and PCPS of the 2D freeform feature FT_{AB} , respectively.

Evidently the PCPS of a freeform feature is always smaller than its CCPS. And in this study, only the PCPS is taken into account during setting up Poisson equation, i.e. the PCPS is Ω described in Section 4.

5.2. Basic definitions and representation of freeform features

Definition 1. Geometry context (GC) of a freeform feature is the difference set between its CCPS and its PCPS.

Essentially, the GC is that control point set in which the control points define the freeform feature with PCPS together but do not need to be repositioned during reusing. In Fig. 2, the control point set $\{P_2, P_3, P_4, P_8, P_9, P_{10}\}$ is the GC of the 2D freeform feature FT_{AB} . Generally, in order to guarantee the reuse quality of a freeform feature, its GCs on the source surface (**source GC**) and the target surface (**target GC**) should be similar to certain extent. As described in Section 3, the similarity between GCs is also decided by degrees, forward differences of knots, etc. So similarly, degree elevation and knot insertion algorithm are used to improve the similarity.

5.2.1. Representation of the freeform feature to be reused

The freeform feature to be reused is represented by a triple $\{BS, PCPS, GC\}$, here BS is the base surface to which the feature belongs, PCPS is its principal control point set and GC is its geometry context on BS.

The first element BS mainly provides the degree, knot vectors and weights of the surface. The PCPS and the GC are two key elements of the freeform feature: PCPS is the actually reused object and GC directly influences the transition region between target surface and the reused feature. When the degrees, knot vectors and weights of two given surface (the source surface and the target surface) are all equal, the PCPS is the unique object which needs to be focused on during reuse. Since it is the minimum control point set which decides the geometry of the feature, therefore the PCPS can effectively improve the efficiency of freeform feature reuse.

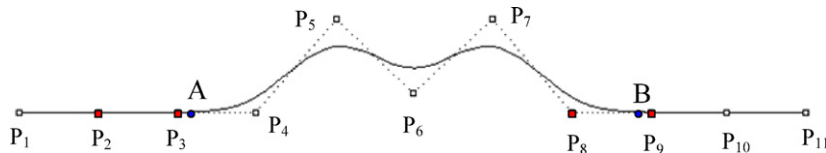


Fig. 2. A freeform feature instance.

6. Reuse of freeform features based on Poisson equation

6.1. Approach overview

In order to achieve a high-quality reuse of freeform features represented by NURBS, we propose a Poisson equation based approach. Here the freeform feature refers to an inner portion of a single NURBS surface with engineering signification. The basic ideas of the approach lie in: realize freeform feature reuse by updating the PCPS on target surface using Poisson equation so that the reused feature can be well fused with the target surface; guided by the conditions for preserving differential properties of reused freeform features, utilize degree elevation and knot insertion operation to achieve the preservation of differential geometry properties; edit the gradient field of PCPS of the source surface during establishing the Poisson equation to make the reused freeform features adapt to the geometry context of the target surface.

The input of our approach is two NURBS surfaces: one is the source surface with a freeform feature; another is the target surface on which the freeform feature will be pasted. Fig. 3 shows the flowchart of the approach. The major steps of the approach are described in more detail below.

6.2. Degree elevation

As mentioned above, the degrees of the source surface and the target surface are supposed to be the same for preserving differential properties and improving the similarity between source GC and target GC. If the two given surfaces do not satisfy this condition, it is necessary to conduct degree elevation on one of them before performing feature reuse.

Let k_1 and k_2 be the degrees of the source surface S_1 and the target surface S_2 respectively, and $k_1 > k_2$, then the degree of S_2 must be raised by $k_1 - k_2$.

The degree elevation algorithm proposed by Cohen et al. [12] is used in our work.

6.3. Identifying of the CCPS and PCPS on source surface

In this work, the CCPS and PCPS on source surface (source CCPS and source PCPS for short) are identified through user interaction, which consists of the following five steps:

- (1) Draw a closed B-Spline contour on a reference plane. Here the reference plane is the tangent plane of a point in the region to be reused on the source surface, which is selected by the user, see Fig. 4(a);

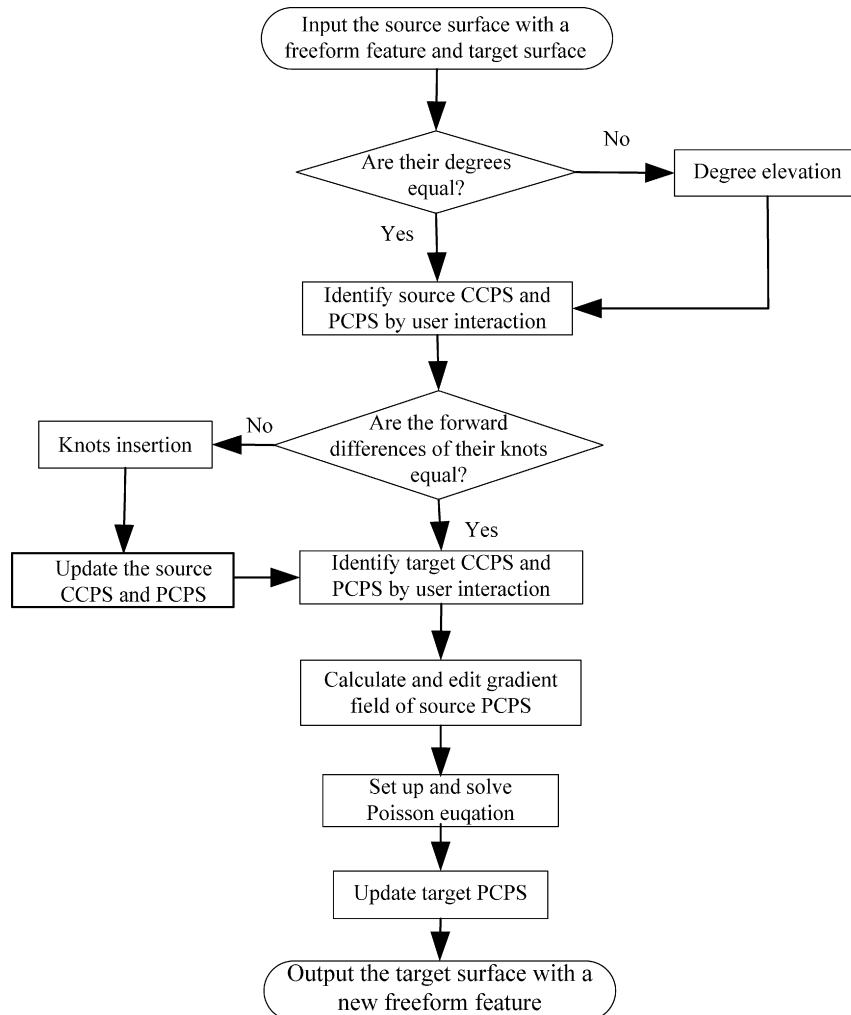


Fig. 3. Flowchart of our approach.

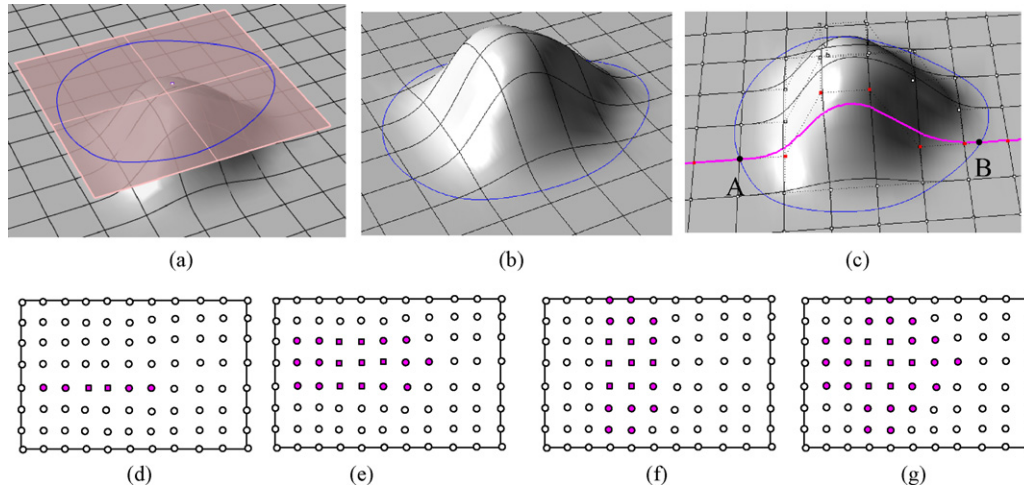


Fig. 4. Identify the source CCPS and source PCPS. (a) The contour on reference plane drawn by the user; (b) the contour projected onto the source surface; (c) an iso-curve defined by a row of control points and its two intersection points with the contour; (d) the CCPS (solid circles) and the PCPS (rectangles) of the row of control points viewed from top; (e) the CCPS (solid circles) and the PCPS (rectangles) of all rows of control points; (f) the CCPS (solid circles) and the PCPS (rectangles) of all columns of control points; (g) the CCPS (solid circles) and the PCPS (rectangles) of the feature.

- (2) Project the contour onto the source surface using the method presented in [13]. The region enclosed by the contour is regarded as the feature to be reused, see Fig. 4(b);
- (3) Identify the CCPS and PCPS of source surface on u direction automatically. For every row of control points of source surface which defines a NURBS curve, if the curve intersects with the contour, the segments of the curve which locates in the region are determined first. Then the CCPS and PCPS of the every curve segment are identified using the method described in Section 5.1. Taking Fig. 4(c) as an example, [AB] is a determined curve segment, and its CCPS and PCPS identified are shown in Fig. 4(d). In Fig. 4(e), all the CCPS and PCPS on u direction are shown.
- (4) The same process in step 3 is conducted for every column of control points, and then the CCPS and PCPS on v direction can be identified, see Fig. 4(f).
- (5) The final source CCPS and PCPS are the union set of CCPS and PCPS on u and v directions, see Fig. 4(g).

6.4. Knot insertion

Source surface and target surface may have different knot vectors, which make the freeform feature reuse difficult and reduce the feature reuse quality. In this work, we adopt knot insertion operation to make the forward differences of the knots of the source and target surfaces equal so that the high-quality feature reuse can be achieved.

The specific algorithm of knot insertion operation consists of two steps: one is determining the locations where the knots should be inserted, and another is inserting the knots located. Chord-

length parameterization must be done before knot insertion operation. The approach consists of the following three steps:

Step 1: Determine the base points.

Because of the local support property of NURBS, the knot insertion operation only performed on the parameter domain of the freeform features. Therefore before determining the locations where the knots are inserted, four base points must be identified first.

- (1) Source control base point and source parameter base point: here we take the control point whose subscripts in u and v directions are both minimums in source CCPS as the **source control base point** and regard the knots in knot vectors which are closest to its Graville abscissa as the **source parameter base point** in the parameter domain.
- (2) Target control base point and target parameter base point: in this work, the target parameter base point is defined on the target surface by user interaction. Suppose that the **target parameter base point** is (u_0, v_0) , two knots $u = u_0$ and $v = v_0$ are inserted. Then a row and a column of control points are inserted simultaneously and the intersection control point is taken as the **target control base point**.

Step 2: Determine the locations where the knots are inserted.

From the parameter base points of source and target surface, the knot insertion operation begins. For every knots of source surface (on both u and v directions), if its forward difference is not equal to that of corresponding knot of source surface, then an appropriate location is marked to insert a knot. And the same process is conducted for the target surface. Taking Fig. 5 as an

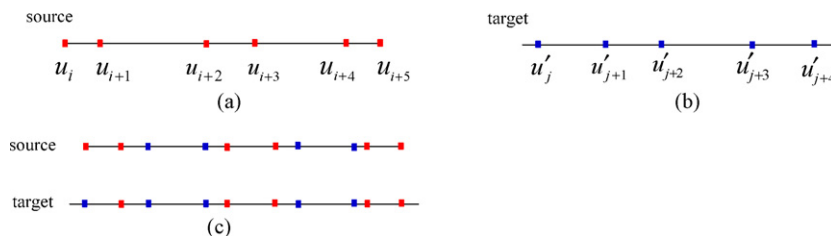


Fig. 5. Knots insertion. (a) The parameter domain of a freeform feature on u direction; (b) a portion of the parameter domain of the target surface on u direction; (c) the result of knot insertion.

example, (a) illustrates the parameter domain of a freeform feature in u direction where u_i is the source parameter base point, and (b) illustrates a portion of the parameter domain of the target surface where u'_j is the target parameter base point. For each knot u_m ($m = i + 1, i + 2, \dots$) in the knot vector on u direction of the source surface, if there is no corresponding knot u'_m satisfying $u'_m - u'_j = u_m - u_i$ in the knot vector on u direction of target surface, then the location u'_m will be marked in the knot vector on u direction of target surface, vice versa. The result of insertion operation is shown in (c).

In some cases, the user would like scale the feature and obtains an appropriate size of it for the target surface. For this purpose, a scale factor α is taken into account while determining a location, i.e. mark a location u'_m which satisfies $u'_m - u'_j = \alpha(u_m - u_i)$ instead of $u'_m - u'_j = u_m - u_i$.

Step 3: Knots insertion.

After the locations where the knots should be inserted are determined, knot insertion operation is performed to insert the knots. Here the widely used method described by Boehm and Prautzsch [14] is adopted to conduct knot insertion of NURBS surfaces.

The knot insertion operation also makes contributions in the following aspects:

- (1) Ensure the number of control points in source CCPS to be equal to that of target CCPS. When a knot is inserted, a row or column of control points will be inserted simultaneously. In this way, one-to-one correspondence can be established between the source CCPS and target CCPS.
- (2) Improve the similarity between the GCs of source surface and target surface. In this case, the knot insertion operation is performed in the parameter domain affected by CCPSs which include GCs. After knot insertion operation, the forward differences of the control points in GCs on source and target surfaces will be equal. And the same forward differences can improve the similarity between GCs from the analysis given in the Appendix A.

After knot insertion operation, the source PCPS and CCPS must be updated using the method described in Section 5.1 because some new control points are inserted during knots inserting.

6.5. Identify the PCPS and CCPS on target surface

During the knot insertion operation, two control base points, the **source control base point** P and the **target control base point** T , are conducted. From these two base points, the CCPS and PCPS on the target surface (target CCPS and target PCPS for short) can be easily identified as follows:

- (1) Find out the relative subscript relationship between P and all control points in source CCPS and PCPS respectively, denoted by R_{ccps} and R_{pcps} :

- (2) Find out all the control points on the target surface whose subscripts satisfy the relationship R_{ccps} and R_{pcps} with T . Those control points comprise the target CCPS and PCPS. The boundary control points of target PCPS can be further determined.

Fig. 6 illustrates the above process.

6.6. Computing and editing gradient field of source PCPS

In our reuse approach, the geometry of the reused feature on the target surface is mainly determined by the Laplacians of the control points in source PCPS (source Laplacians), which can be easily computed using Eq. (6). However, in some cases the source Laplacians may not guarantee the quality of feature reuse such as the posture of the target surface is not very consistent with that of the source surface (e.g., their orientations are different). In order to achieve an expected feature reuse effect, the source Laplacians need to be further edited.

6.6.1. Laplacian rotation

When the orientations of the source GC and target GC are different, Laplacian rotation operation must be performed, which includes the following two steps:

- (1) *Setting up of the rotation matrix*

The local rotation matrix R_p at control point P is taken from the local rotation between their corresponding normals in source and target surfaces. In Fig. 7, let S be the control net of source surface and \tilde{S} be a smooth version of S . The control net \tilde{S} is a low-frequency net associated with S , which can be generated by filtering [15]. By regarding the control net \tilde{S} as a mesh, the normal of each control point can be calculated with the usual method of computing normals for mesh. Suppose P is a control point in S , P' is its counterpart in \tilde{S} , and \vec{p} is the normal of P' , then the normal of P is set to \vec{p} . The same method is applied to target surface to calculate all the normals of the control points in target PCPS. Let T be the counterpart of P in target PCPS, and \vec{t} be its normal, then the rotation matrix R_p of P is obtained by rotating \vec{p} to \vec{t} around P .

- (2) *Rotation*

Based on the rotation matrix determined above, for each control point P in source PCPS, its Laplacian is recalculated as follows:

- (a) Update its N_p by rotating every control point in N_p with the same rotation matrix R_p and
- (b) Calculate the new Laplacian of P using Eq. (6) using the new N_p .

Taking Fig. 8 as an example, where the Laplacian of the control point P needs to be recalculated, and P_1, P_2, P_3, P_4 compose its N_p , \vec{p} is the normal of P and \vec{t} is the normal of its

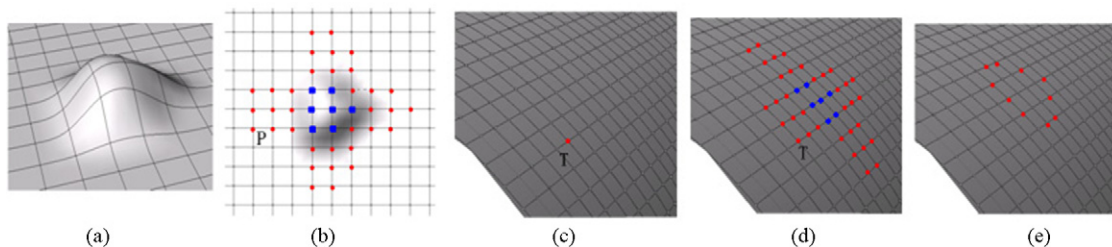


Fig. 6. Identification of the target CCPS and PCPS. (a) The feature to be reused; (b) the source PCPS (rectangles), source GC (solid circles) and source control base point P ; (c) the target control base point T ; (d) the target PCPS (rectangles) and GC (solid circles); (e) The boundary of the target PCPS (solid circles).

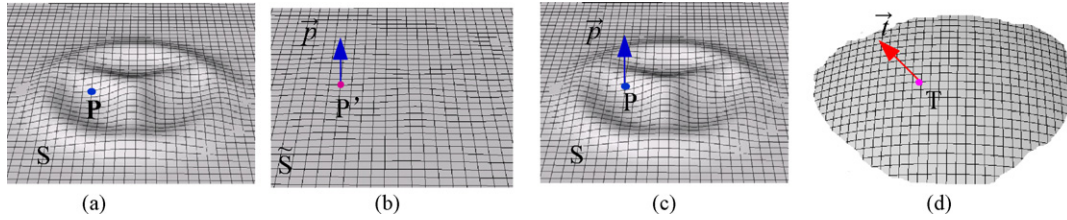


Fig. 7. Setting up the rotation matrix of P in the source PCPS. (a) The control net of the source surface and P ; (b) a smooth version of control net of source surface and the normal of P' , denoted by \vec{p} ; (c) the normal of P is assigned to \vec{p} ; (d) the normal of T , denoted by \vec{t} .

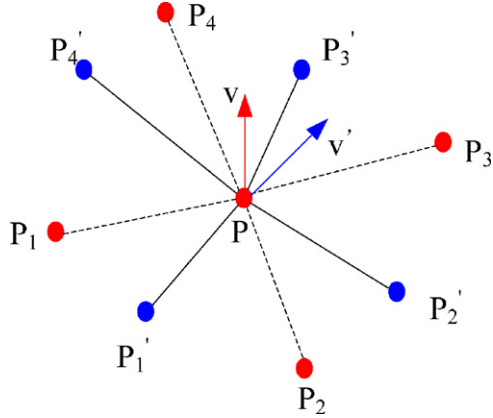


Fig. 8. Laplacian editing of the control point P using local rotation.

corresponding control point on target surface. After applying the rotation transformation to P_1, P_2, P_3 , and P_4, P_1', P_2', P_3' , and P_4' are obtained, and then the Laplacian of P becomes:

$$\Delta f_p = \sum f_{pq} = 4P - \sum_{i=1}^4 P'_i \quad (7)$$

6.6.2. Laplacian editing for preserving original feature on target surface

In Section 4.2, we regard the gradient field of the source PCPS as the guidance of the Poisson equation and it is usually leads to a desirable reuse effect. However, there are situations requiring special adaption to target surface. Taking Fig. 9 as an example, there is a Benz logo on a plane (a) which will be pasted on a bended surface (b). When a rectangle region on the plane (red rectangle in (a)) is chosen to be reused, an unsatisfactory result (c) will be obtained using directly the gradient field of source surface which only reflects the shape of source surface and neglect the shape of target surface. One solution to this problem is to combine the gradient field of the source surface with that of target surface to get a desirable guidance field which is suitable to the target surface. Consider that the feature always has bigger gradient, so we preserve the stronger of gradient in source surface and that in target surface using the following formula:

$$v(x) = \begin{cases} \nabla f(x), & \text{if } (|\nabla f(x)| > |\nabla g(x)|) \\ \nabla g(x), & \text{otherwise} \end{cases} \quad (8)$$

And the discrete counterpart of this guidance field is

$$v_{pq} = \begin{cases} g_s - g_t, & \text{if } (|g_s - g_t| > |f_p - f_q|) \\ f_p - f_q, & \text{otherwise} \end{cases} \quad (9)$$

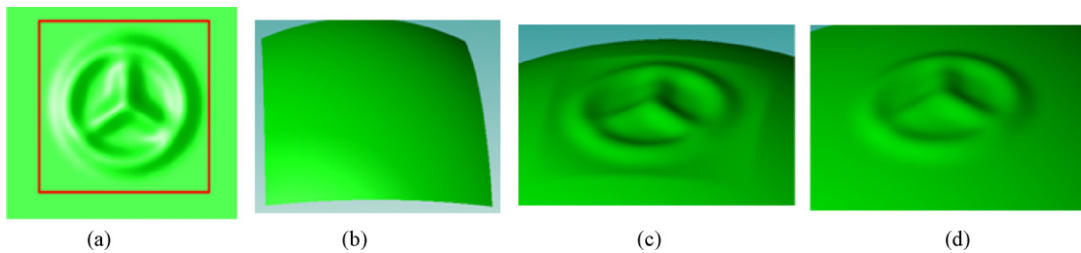


Fig. 9. Laplacian editing for preserving the properties of the target surface. (a) A Benz logo on a plane; (b) a bended surface regarded as a target surface; (c) the reuse result using the original guidance field; (d) the reuse result using the modified guidance field.

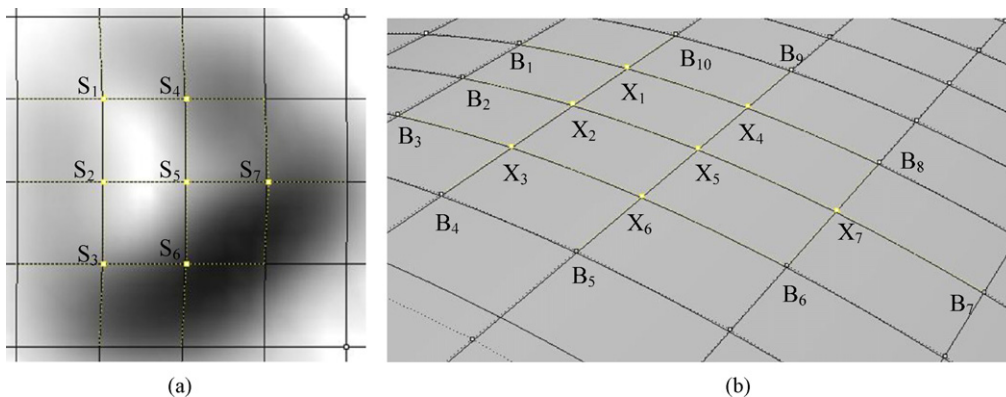


Fig. 10. An instance of setting up Poisson equation. (a) The source PCPS; (b) the target PCPS and its boundary.

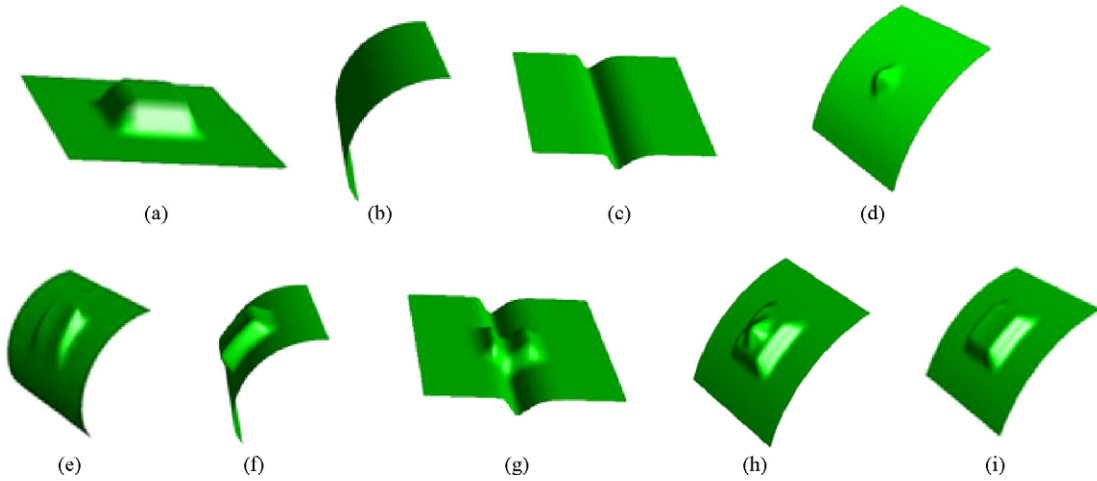


Fig. 11. Reuse examples of a protrusion feature. (a) The source surface with a protrusion feature; (b–d) are three target surfaces; (e) and (f) protrusion feature is reused on target surface (b); (g) protrusion feature is reused on target surface (c); (h) and (i) protrusion feature is reused on target surface (d).

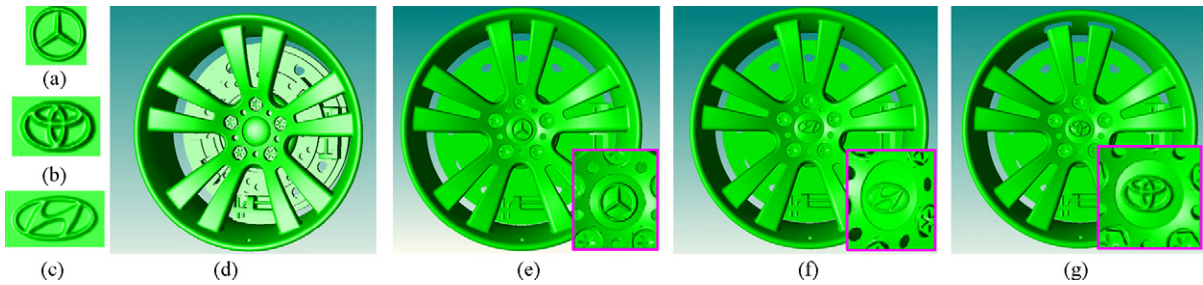


Fig. 12. The logos of BENZ, TOYOTA and HYUNDAI are, respectively reused on a wheel. (a–c) BENZ, TOYOTA and HYUNDAI logos; (d–j) feature reuse results.

where p and q are control points in source PCPS, s and t are their counterparts in target PCPS. The desired result is shown in Fig. 9(d).

6.7. Set up and solve Poisson equation

After computing the source Laplacians and determining the boundary of target PCPS, the Poisson can be set up. For every control point in target PCPS, an Eq. (5) can be set up and all the equations construct a classical sparse linear system, namely Poisson equation.

Taking Fig. 10 as an example, where $S = \{S_1, S_2, \dots, S_7\}$ in (a) is the source PCPS, $X = \{X_1, X_2, \dots, X_7\}$ in (b) is the target PCPS, and $B = \{B_1, B_2, \dots, B_{10}\}$ in (b) is the boundary of X . Suppose $x = \{x_1, x_2,$

$\dots, x_7\}$ is the x coordinate set of X , and $b = \{b_1, b_2, \dots, b_{10}\}$ is the x coordinate set of B , $\Delta(S_i)$ ($i = 1, 2, \dots, 7$) is the Laplacian of the control point S_i related to x coordinate and it can be calculated using Eq. (6) and further edited using Eqs. (7) and (9). Then the Poisson equation related to x coordinate is

$$\begin{cases} 4x_1 - x_2 - x_4 = b_1 + b_{10} + \Delta(S_1) \\ 4x_2 - x_1 - x_3 - x_5 = b_5 + \Delta(S_2) \\ 4x_3 - x_2 - x_6 = b_3 + b_4 + \Delta(S_3) \\ 4x_4 - x_1 - x_5 = b_9 + b_8 + \Delta(S_4) \\ 4x_5 - x_2 - x_6 - x_7 - x_4 = \Delta(S_5) \\ 4x_6 - x_3 - x_5 = b_5 + b_6 + \Delta(S_6) \\ 4x_7 - x_5 = b_6 + b_7 + b_8 + \Delta(S_7) \end{cases} \quad (10)$$

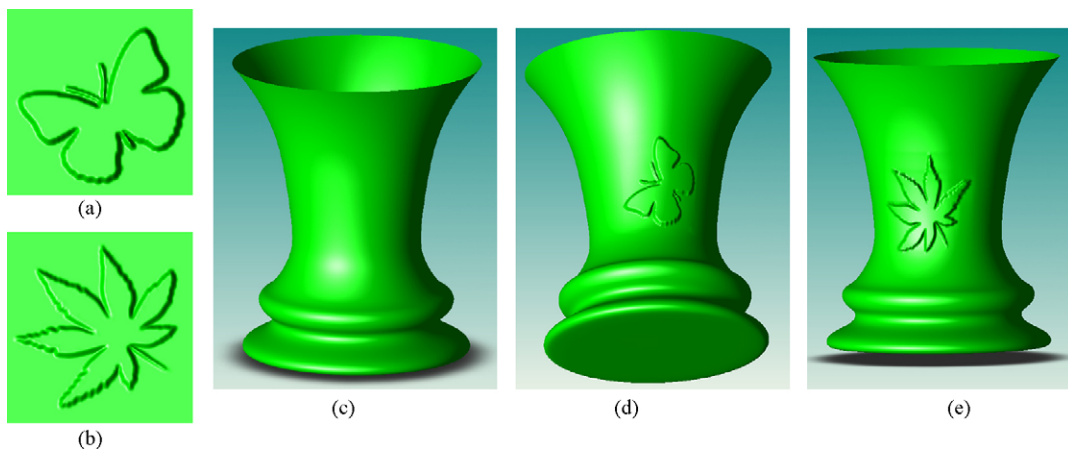


Fig. 13. A butterfly and a maple leaf are reused on a vase. (a) A butterfly model; (b) a maple leaf model; (c) a vase model; (d) the vase with a butterfly reused; (e) the vase with a maple leaf reused.

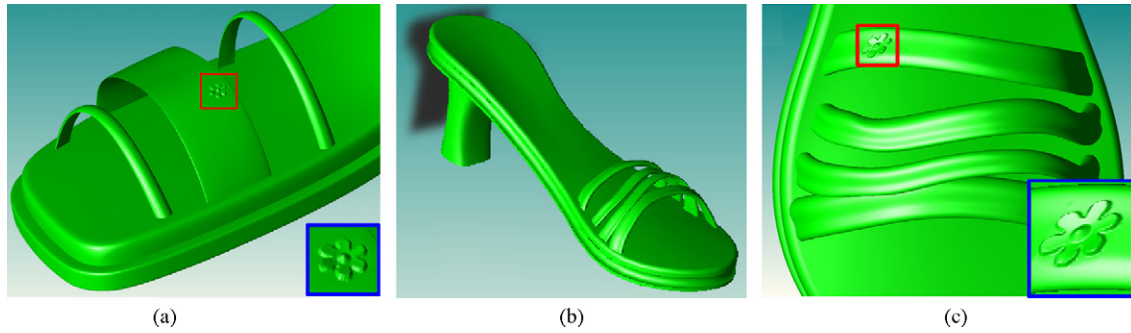


Fig. 14. An ornamental flower is reused on a female shoe. (a) A shoe model with an ornamental flower; (b) another female shoe model; (c) the feature reuse result.

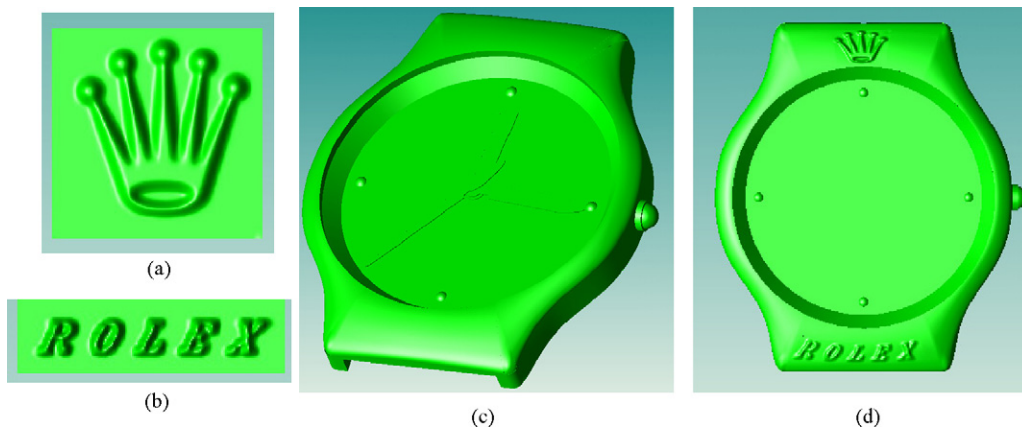


Fig. 15. The Rolex logo model and the text "ROLEX" model are reused on a watch model. (a) The Rolex logo model; (b) the text "ROLEX" model; (c) a watch model; (d) the feature reuse result.

The Poisson equation related to y or z is similar to Eq. (10), and the only difference is that the x coordinate is replaced with y or z coordinate.

By solving Poisson equations, the new 3D coordinates of each control point in target $PCPS$ are obtained, and after the coordinates are updated, the reuse of the freeform feature is realized.

7. Implementation

The method presented in this study has been implemented with Visual C++7.1 based on the geometric modeling engine ACIS 13 [16] and HOOPS 10 [17]. A number of test examples as well as the test results are shown in Figs. 11–16.

Fig. 11(a) shows a source surface with a protrusion feature, (b–d) are three target surface, (e–i) show the reused results.

Among above results, the result of feature reuse as shown in (f) is achieved by conducting the Laplacian editing based on local rotation (Eq. (7)). In contrast, the result of the feature reuse where the Laplacian is not edited is shown in (e), which is obviously not good. This is because the latter does not deal with the posture inconsistency between the source and target surface. In view that there is already a groove feature within the pasted area on the target surface shown in (c) and the user wants to preserve the groove after reuse, the Laplacian is edited via Eq. (9) and the result shown in (g) is obtained. In order to achieve the result shown in (h), the Laplacian is edited not only using Eq. (7) but using Eq. (9). In

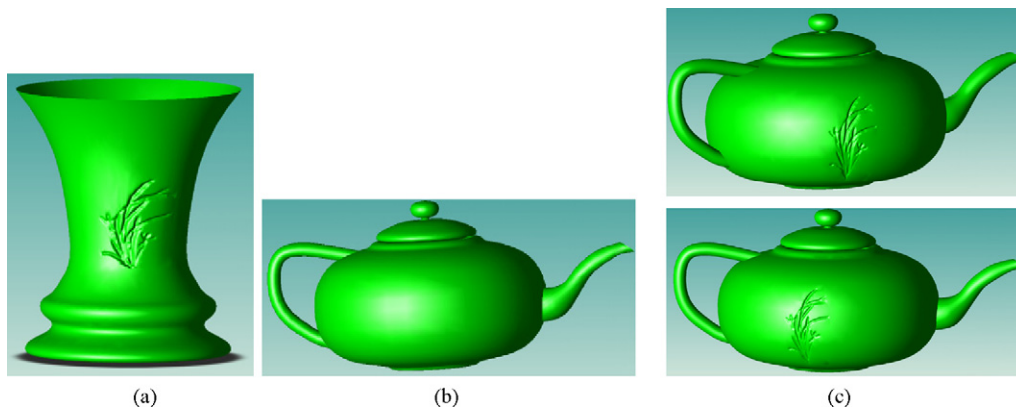


Fig. 16. An orchid feature is reused on a teapot model. (a) A vase model with an orchid feature; (b) a teapot model; (c) the feature reuse results.

Table 1
Comparison between different algorithms of freeform feature reuse.

Method	Representation of resultant surface	Continuity	Preserve feature shape or not
Banghiel's	Composite surface	Bad	Inexactly
Wang's	Composite implicit surface	Bad	Exactly, but a transitional surface is added
Our method	A single NUBRS surface	Keep the continuity of target surface good	The feature is adapted to the shape of the target surface

contrast, the result of the feature reuse where the Laplacians is only edited by rotation transformation is shown in (i).

In Figs. 12–16, several more complex freeform feature reuse examples are shown.

8. Conclusions

In this paper, a novel approach for reusing the freeform features represented by NURBS is proposed. The characteristics of this method consist of the following three aspects:

- (1) Conditions for preserving differential properties of reused freeform features are derived and used to guide the approach to achieve the preservation of the differential geometry properties.
- (2) Freeform feature reuse is achieved by updating the target PCPS using Poisson equation. In this way, the reused feature can be well fused with and adapt to the geometry context of the target surface, and the target surface with the reused feature is still a single NURBS.
- (3) A reuse-oriented representation of freeform features is put forward, which consists of the principal control point set, geometry context and basic surface of the feature. With this representation, Poisson equation is limited on the PCPS and thus improves the efficiency of freeform feature reuse.

The comparison between our method and other closely related approaches is given in Table 1.

Our future work will focus on the following aspects: extend the approach to achieve the reuse of the freeform features with trimmed surfaces; enable the approach to deal with the cases that the reused freeform features are rotated.

Acknowledgments

The authors are very grateful to the financial support from NSF of China (Nos. 60736019 and 60574061), 863 High Tech. Plan of China (2007AA04Z149), and Zhejiang Provincial Natural Science Foundation of China (Z107497).

Appendix A. Derivation of conditions for preserving differential geometry properties of reused freeform features

As the first step, we derive the conditions guaranteeing the derivatives of two B-Spline curves equal.

The p -th B-Spline curve can be defined as

$$C(u) = \sum_{i=0}^n N_{i,p}(u) P_i \quad (A1)$$

where $\{P_i\}$ are the control points, and $\{N_{i,p}(u)\}$ are the p -th degree B-Spline basis functions defined on the knot vector $U = \{u_0, u_1, \dots,$

$u_{n+p=1}\}$. The $N_{i,p}(u)$ is defined as

$$N_{i,p} = \begin{cases} N_{i,0}(u) = \begin{cases} 1, & \text{if } (u_i \leq u \leq u_{i+1}) \\ 0, & \text{otherwise} \end{cases} \\ \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \end{cases} \quad (A2)$$

The k -th derivatives of $C(u)$ is

$$C^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k}(u) P_i^{(k)} \quad (A3)$$

with

$$P_i^{(k)} = \begin{cases} P_i, & k = 0 \\ \frac{p - k + 1}{u_{i+p+1} - u_{i+k}} (P_{i+1}^{(k-1)} - P_i^{(k-1)}), & k > 0 \end{cases} \quad (A4)$$

Here the forward difference $P_{i+1}^{(k-1)} - P_i^{(k-1)}$, denoted by $\nabla P_i^{(k)}$, can be expressed by $\sum \pm (P_{j+1} - P_j)$ using recursive relationship. For example:

$$\begin{aligned} \nabla P_i^{(1)} &= P_{i+1} - P_i \\ \nabla P_i^{(2)} &= P_{i+1}^{(1)} - P_i^{(1)} = (P_{i+2} - P_{i+1}) - (P_{i+1} - P_i) \\ \nabla P_i^{(3)} &= [(P_{i+3} - P_{i+2}) - (P_{i+2} - P_{i+1})] - [(P_{i+2} - P_{i+1}) - (P_{i+1} - P_i)] \end{aligned}$$

So, in order to preserve the k -th ($k \leq p$) derivatives, the forward difference of control points must be unchanged. Analogously, by analyzing the denominator of Eq. (4) we can get another condition: the forward difference of knots must be kept unchangeable (when $k = p$). Based on these, we can obtain the following conclusion:

Conclusion 1: For two p -th degree B-Spline curves, their k -th ($k \leq p$) derivatives are equal if

- (1) their forward differences of control points ∇P_i are equal;
- (2) their forward differences of knots ∇u_i are equal.

Based on the conclusion 1, we can further derive the conditions guaranteeing the derivatives of two NURBS curves equal as follows.

NURBS curve is a rational B-Spline curve, and p -th degree NURBS curve can be written as

$$C(u) = \frac{\omega(u)C(u)}{\omega(u)} = \frac{A(u)}{\omega(u)} \quad (5)$$

where $A(u) = \sum_{i=0}^n N_{i,p}(u) \omega_i P_i$, $\omega(u) = \sum_{i=0}^n N_{i,p}(u) \omega_i$. It is clear that the $A(u)$ and $\omega(u)$ are all B-Spline curves, and their differences from Eq. (1) are that their control points are weighted control points ($\omega_i P_i$) and weights (ω_i), respectively. Analogously we get the following conclusion:

Conclusion 2: For two p -th degree NURBS curves, their k -th derivatives are equal if

- (1) their forward differences of weighted control points $\nabla \omega_i P_i$ are equal;
- (2) their forward differences of knots ∇u_i are equal;
- (3) their forward differences of weights $\nabla \omega_i$ are equal.

We can derive the conditions for guaranteeing the derivatives of the NURBS surface equal easily from the condition 2.

The NURBS surface is a tensor product B-Spline surface and its derivatives can be computed by twice derivative computations of NURBS curve. For a NURBS surface with $(m+1) \times (n+1)$ control points, $(m+1)$ partial derivatives are computed first along u direction

with fixed v ; next, regarding the $(m+1)$ partial derivatives as intermediate control points, the final derivatives can be computed. So the partial derivative of a NURBS surface is also determined by the forward differences of weighted control points, knots and weights, but the differences of weighted control points and weights are bidirectional, i.e. $\nabla\omega_{ij}P_{i,j} = ((\omega_{i+1,j}P_{i+1,j} - \omega_{i,j}P_{i,j}), (\omega_{i,j+1}P_{i,j+1} - \omega_{i,j}P_{i,j}))$, $\nabla\omega_{i,j} = ((\omega_{i+1,j} - \omega_{i,j}), (\omega_{i,j+1} - \omega_{i,j}))$. Then, we get the following conclusion:

Conclusion 3: For two NURBS surfaces, $S(u, v) = \sum \sum N_{i,p}(u)N_{j,q}(v)P_{i,j}$ with knot vector $\{u_i, v_i\}$ and weights $\{\omega_{i,j}\}$, $S'(u, v) = \sum \sum N_{i,m}(u)N_{j,n}(v)P'_{i,j}$ with knot vector $\{u'_i, v'_i\}$ and weights $\{\omega'_{i,j}\}$, their $(k+l)$ -th derivatives are equal if

- (1) Their degrees in both directions are equal, i.e. $p = m$ and $q = n$;
- (2) Their forward differences of weighted control points are equal, i.e. $\nabla\omega_{i,j}P_{i,j} = \nabla\omega'_{i,j}P'_{i,j}$;
- (3) Their forward differences of weights are equal, i.e. $\nabla\omega_{i,j} = \nabla\omega'_{i,j}$;
- (4) The forward difference of knots in both directions are equal, i.e. $\nabla u_i = \nabla u'_i$ and $\nabla v_i = \nabla v'_i$.

References

- [1] J.J. Shah, M. Mäntylä, Parametric and Feature-Based CAD/CAM, Wiley, New York, 1995.
- [2] H. Biermann, I. Martin, F. Bernardini, D. Zorin, Cut-and-Paste editing of multi-resolution surface, in: Processing of ACM SIGGRAPH, July 21–26, San Antonio, TX, (2002), pp. 312–321.
- [3] C. Dumont, D.R. Wallace, FreeForm surface copy-and-paste: a mechanism for transferring brand-identity elements from one object to another, in: Processing of DETC 2003 ASME Design Engineering Technical Conference and Computer and Information in Engineering Conference, September 2–6, Chicago, IL, USA, 2003.
- [4] Y. Furukawa, H. Masuda, Cut-and-Paste editing based on constrained B-spline volume fitting, Computer Graphics International (2003) (short paper).
- [5] Y.Z. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, et al., Mesh editing with Poisson-based gradient field manipulation, in: Processing of ACM SIGGRAPH, August 8–12, Los Angeles, CA, (2004), pp. 644–651.
- [6] C.S. Wang, J.S.M. Vergeest, I. Horvath, Cross model shape reuse: copying and pasting of free form feature, in: Processing of DETC 2002 ASME Design Engineering Technical Conference and Computer and Information in Engineering Conference, 29 September–2 October, Montreal, Canada, 2002.
- [7] C. Barghiel, R. Bartels, D. Forsey, Pasting spline surfaces, in: Mathematical Method for Curves and Surfaces, Ulnik, Norway, 1994, pp. 31–40.
- [8] B. Conrad, S. Mann, Better pasting via quasi-interpolation, in: Curves and Surfaces Design, SaintMalo, 1999, pp. 26–37.
- [9] E. van den Berg, R. Bidara, W.F. Bronsvort, Construction of freeform feature models with attachments, in: Processing of DETC 2004 ASME Design Engineering Technical Conference, 28 September–2 October, Salt Lake City, 2004.
- [10] W. Zhao, S. Gao, Y. Liu, Poisson-based reuse of freeform feature, in: Processing of DETC 2006 ASME Design Engineering Technical Conference and Computer and Information in Engineering Conference, September 10–13, Philadelphia, USA, 2006.
- [11] P.M. Pérez, A. Blake, Poisson image editing, in: Processing of ACM SIGGRAPH, July 27–31, San Diego, CA, (2003), pp. 313–318.
- [12] E. Cohen, T. Lyche, L.L. Schumaker, Algorithms for degree raising of splines, ACM Transactions on Graphics 4 (5) (1985) 171–181.
- [13] L.A. Piegl, W. Tiller, The NURBS Book, Springer Press, New York, 1995.
- [14] W. Boehm, H. Prautzsch, The Insertion algorithm, CAD 17 (2) (1985) 58–59.
- [15] M. Desbrun, M. Meyer, P. Schröder, H. Barra, Implicit fairing of irregular meshes using diffusion and curvature flow, in: Processing of ACM SIGGRAPH, August 8–13, Los Angeles, (1999), pp. 317–324.
- [16] <http://www.spatial.com>.
- [17] <http://www.hoops3d.com>.



Wei Zhao is a Ph.D candidate of the State Key Lab. of CAD&CG, Zhejiang University. He received his B.S degree (2003) in computer science from the Shan Dong University of Science and Technology. His research interests include freeform feature techniques, computer graphics and geometry modeling etc.



Dr. Shuming Gao is a Professor in State Key Lab of CAD&CG, Zhejiang University. He received his Ph.D degree from the Applied Mathematics Department of Zhejiang University in 1990, and was a visiting scholar and a visiting Professor in the Design Automation Lab of Arizona State University, respectively in 1996 and 2001. His research interests include product modeling, CAX integration, collaborative design, virtual reality in design and manufacturing, MEMS CAD, etc.



Dr. Liu Yusheng is currently a Professor in State Key Lab. of CAD&CG, Zhejiang University, PR China. He received his Ph.D. degree in Mechanical Engineering from Zhejiang Univ., PR China in 2000. After that, he continued his postdoctoral research in State Key Lab. of CAD&CG, Zhejiang Univ. and City Univ. of Hongkong in 2000–2003. His main research interests are CAD/CAM, 3D model retrieval, computer aided tolerancing and virtual prototyping.



Dr. Hongwei Lin is an Associate Professor in State Key Laboratory of CAD&CG, Zhejiang University, China. He received his B.Sc. from Department of Applied Mathematics at Zhejiang University in 1996, and Ph.D. from Department of Mathematics at Zhejiang University in 2004. He worked as a communication engineer from 1996 to 1999. His current research interests are in computer aided geometric design, computer graphics, and image process.