International Journal of Image and Graphics
Vol. 10, No. 1 (2010) 41–56

CONSTRUCTING SURFACE FEATURES THROUGH DEFORMATION

XIAOPING WANG*, SHENGLAN LIU and LIYAN ZHANG
CAD/CAM Engineering Research Center
Nanjing University of Aeronautics and Astronautics
Nanjing 210016, The People's Republic of China
**harayano@hotmail.com*

Received 5 April 2007
Revised 30 October 2008
Accepted 9 August 2009

A frequent problem in computer aided mechanical design is the construction of arbitrarily-shaped ribs and beads on surfaces, to increase their rigidity or for aesthetic reasons. We improve upon a previous mathematical approach for defining such ribs and beads, based on using so-called extension functions to define a deformation matrix, which is then applied to the underlying shape. Our improvements offer important practical advantages: firstly, by use of cosine extension functions, we get a greater control over, and flexibility of, rib shape, including the possibility of repeating ribs; secondly, we can directly control the spine curves. We give experimental results to demonstrate that the method is simple and intuitive, has low computational cost, and is potentially useful for computer aided design, computer graphics and other applications.

Keywords: Parametric surface; deformation; surface curve; interpolation; G^2 continuity; rib.

1. Introduction

Currently, using computer aided design (CAD) systems design of surface features such as beads and ribs on workpieces, e.g. sheet metal objects, requires a significant effort. We propose a new method for designing beads and ribs using a surface shape modification approach. Shape modification has been widely studied previously. For example, Piegl[1] proposed two methods for modifying the shape of non uniform rational B-spline (NURBS) surfaces by moving the control points and changing the weight factors. Ma[2] put forward an approach to update a local area of a B-spline surface based on a set of locally distributed and unorganized points in three-dimensional space. Ishida[3] presented a method that can produce an arbitrary direct modification of a surface using displacement functions. Georgiades[4] designed interactive tools for locally editing a surface by altering its intrinsic geometric measures; this approach is independent of the underlying surface representation (parametric, implicit, algebraic, etc). Singh[5] gave a geometric deformation technique based on the use of control wires. Léon[6,7] described a method of surface deformation based on mechanical equilibrium of a bar network, based on concepts of fixed and free nodes, and internal and external forces acting on a surface's control mesh. Hu[8] considered shape modification of NURBS surfaces with geometric constraints leading to a problem of constrained optimization and energy minimization. Mari[9] presented an approach which enabless the user to easily characterize and control the shape defined to a closed surface.

These methods, and others not mentioned here, possess varying advantages and have varying practical importance. However, using these methods, it is not very easy for the user to create ribs or beads on a surface for various reasons: ribs or beads are often required to be repeated at intervals across the surface in practice; furthermore, the demands for precise location and control over deformation are rather higher than those usually encountered for shape modification. We thus give a new method which takes these issues into account.

Taking as inspiration the principle of sheet metal forming through stamping, we[10] previously proposed a method of deformation based on a so-called *extension function*. However the method is not entirely satisfactory, and has two main drawbacks. Firstly, extension functions can only generate a limited range of feature shapes. Secondly, the method does not directly determine the so-called *spine* (or bounding curve), which is often required in practice. The new method in this paper makes improves on our earlier work by redefining the extension function and introducing a design technique based on spines. This method involves three steps: firstly, we construct extension functions based on cosine functions; secondly, based on the extension function, we create a deformation matrix; finally, we apply the deformation matrix to the surface, to produce the final surface. The new extension functions have good geometric and analytic properties, and contain several variable parameters, allowing the deformation matrix constructed from them to have analogous properties to a die, of controllable shape. The method can be used to create ribs on any parametric surface. Our implementation demonstrates that the method is capable of producing repeated ribs where required, and can meet the demands of precise location and control of deformation. Moreover, user control is very simple and intuitive.

The rest of the paper is organized as follows. Section 2 explains our mathematical model; the basic concepts, the deformation formula and its control, and complex deformation, are discussed in Secs. 2.1, 2.2 and 2.3, respectively. Determination of spine is discussed in Sec. 3, as are the steps used in deformation. Practical examples are given in Sec. 4. Finally, Sec. 5 gives conclusions and further research directions.

2. Mathematical Model

Here we give a deformation model whose ideas are guided by sheet metal forming by stamping, as used in manufacturing industry. firstly, we construct extension functions; secondly, with the extension function, we create a deformation matrix (conceptually corresponding to a die); finally, we apply the deformation matrix to the surface (conceptually corresponding to a stamping process), to produce the final surface. Since several control parameters are introduce into extension functions and hence into the deformation matrix, diversified deformation effects can be created by changing these parameters (conceptually corresponding to using different dies).

2.1. *Extension function and related concepts*

Our goals are to produce *ribs* on a parametric surface with precisely controlled location, which are smooth, and join smoothly to the rest of the surface, which can be controlled interactively by simple means, and which may be repeated at intervals across the surface if desired. We define the following function.

Let $C : \varphi(u,v) = A$ be a continuous curve in the (u,v) plane, where $A \neq 0$. Let $U = \{(u,v)||\varphi(u,v) - A| \leq S, 0 < S \in R\}$ be the region bounded by C. We suppose the function $\varphi(u,v)$ to be continuous, with continuous partial derivatives of degree up to $(2n-1)$ on the boundary of U, where n is a natural number. Then the composite function

$$E(u,v) = E(u,v,h,n,\omega) = \begin{cases} 1 + h(1 + (-1)^{\omega-1}\cos(\omega\pi(\varphi(u,v) - A)/S))^n, & |\varphi(u,v) - A| \leq S \\ 1, & |\varphi(u,v) - A| > S \end{cases}$$

is called a ***cosine extension function,*** where ω and n are positive integers, and A and h are real values. We use the following terminology: the curve C is called the ***spine***, U is the ***support region***, the boundary of U is the ***bounding curve***, ω is the ***repeat count***, n is the ***smoothness degree***, and h is the ***magnitude of variation.***

The geometric meaning of these values is as follows: the spine describes in general terms the geometric path of the deformation (rib) on the surface, S controls the size of the deformation region, and ω affects the number of repeated ribs, as shown in the examples in Figs. 6–12. The other control parameters h, n also affect the shape of the deformation. In particular, h need not be a constant, but can be for example a function of u and v, giving greater control over deformation. It is easy to prove in the latter case that the continuity of the deformed surface across the bounding curve remains the same as in the case where h is constant.

The logical relations between our concepts used for deformation are illustrated in Fig. 1.

We note that extension functions possess the following properties:

1. If ω is an odd integer, then

$$E(u,v)|_{\varphi(u,v)=\pm\frac{(2j-1)S}{\omega}+A} = 1$$

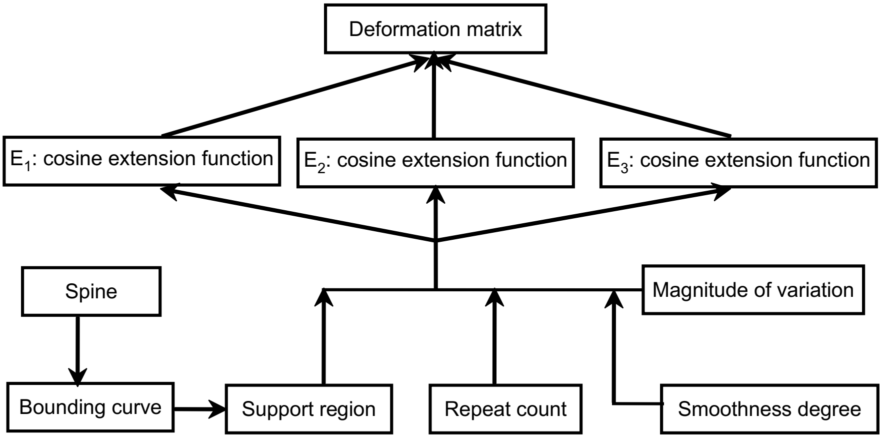

Fig. 1. Conceptual diagrams of the deformation systems.

and

$$\left.\frac{\partial^i E(u,v)}{\partial u^k \partial v^l}\right|_{\varphi(u,v)=\pm\frac{(2j-1)S}{\omega}+A} = 0,$$

where

$$j = 1, 2, \ldots, (\omega+1)/2, \quad 0 \le i = k+l, k, \;\; l \le 2n-1.$$

If ω is an even integer, then

$$E(u,v)|_{\varphi(u,v)=\pm\frac{2jS}{\omega}+A} = 1$$

and

$$\left.\frac{\partial^i E(u,v)}{\partial u^k \partial v^l}\right|_{\varphi(u,v)=\pm\frac{2jS}{\omega}+A} = 0,$$

where

$$j = 0, 1, \ldots, \omega/2, \quad 0 \le i = k+l, k, \;\; l \le 2n-1.$$

2. When ω is an odd integer, it attains its extreme value on the curves

$$\varphi(u,v) = \pm\frac{2jS}{\omega} + A,$$

and furthermore,

$$\left.\frac{\partial^i E(u,v)}{\partial u^k \partial v^l}\right|_{\varphi(u,v)=\pm\frac{2jS}{\omega}+A} = 0$$

where

$$j = 0, 1, \ldots, (\omega-1)/2, \quad i = k+l, k, \;\; l \in N.$$

When ω is an even integer, it attains its extreme value on the curves

$$\varphi(u,v) = \pm\frac{(2j-1)S}{\omega} + A,$$

and furthermore,

$$\left.\frac{\partial^i E(u,v)}{\partial u^k \partial v^l}\right|_{\varphi(u,v)=\pm\frac{(2j-1)S}{\omega}+A} = 0,$$

where

$$j = 1, 2, \ldots, \omega/2, \quad i = k + l, k, \quad l \in N.$$

3. They are periodic in that

$$E(u,v)|_{\varphi(u,v)=t\pm\frac{2S}{\omega}-A} = E(u,v)|_t, \quad A - S \leq t, t \pm \frac{2S}{\omega} - A \leq A + S.$$

The straightforward proofs of the above properties are omitted.

2.2. *Formula for basic deformation*

To apply a deformation, we choose a point O′ as the so-called *deformation center*. There are two reasons for doing so. The first is to make the deformation independent of the coordinate system. The second is that such a center can be used to further control the deformation.

Let $\boldsymbol{p}(u,v) = (x(u,v), y(u,v), z(u,v))^T$ (here, T denotes transpose) be a C^r surface defined on the domain Ω, where $\Omega \subset R^2$. Let $E_i(u,v) = E(u,v,h_i,n,\omega_i)$ be three extension functions whose support region U belongs to Ω, where $2n - 1 \leq r$, $i = 1,2,3$. Now let D be a diagonal matrix defined by $D = diag\,(E_1(u,v), E_2(u,v), E_3(u,v))$; we call D the ***deformation matrix***.

We obtain the deformed surface $\boldsymbol{p}_d(u,v)$ from the original surface $\boldsymbol{p}(u,v)$ using the following formula:

$$\boldsymbol{p}_d(u,v) = D(\boldsymbol{p}(u,v) - \boldsymbol{O}') + \boldsymbol{O}', \quad (u,v) \in \Omega. \tag{1}$$

Taking partial derivatives of (1) and using the definition of the extension function, one can easily prove that the deformed surface $\boldsymbol{p}_d(u,v)$ possesses continuity of order $(2n-1)$ on the bounding curve.

To give the reader further insight into the way in which the parameters affect the nature of the deformation produced, we note the following:

(1) Changing the sign of h_i affects whether the rib is a depression into the surface, or a protrusion on the surface. Compare Fig. 7 with Fig. 8, and Fig. 9 with Fig. 10.
(2) The magnitude of h_i affects the height of the rib with respect to the underlying surface.

(3) Changing n affects the degree of the surface's continuity where it meets the rib. Furthermore, increasing n results in the rib following the original surface more closely near the bounding curve.
(4) The spine C controls the underlying path of the rib across the surface.
(5) Increasing the repeat count ω_i allows multiple adjacent ribs of similar shape to be created on the surface, as shown in Figs. 4, 8 and 9.
(6) The constant S, allows control over the rib's width.

These parameters can be varied independently to provide easy control over the nature of the deformation produced.

2.3. *Fomula for complex deformation*

One can simultaneously apply several repeated deformations with different spines, over different regions, to get more complex deformation effects, as shown in Figs. 4, 5, 9, 11 and 12.

Let D_i be several deformation matrices constructed using various cosine extension functions. Then the overall deformation of the surface $\boldsymbol{p}(u,v)$ is given by:

$$\boldsymbol{p}_d(u,v) = \left(\prod_{i=1}^{m} D_i\right)(\boldsymbol{p}(u,v) - \boldsymbol{O}') + \boldsymbol{O}', \quad (u,v) \in \Omega, \ \ i = 1, 2, \ldots, m. \tag{2}$$

Furthermore, if the various regions used to define the extension functions intersect: $\bar{U} = \bigcap_{k=1}^{s} U_{i_k} \neq \phi$, where U_{i_k} is surrounded by boundary curves $B_{i_k,j}$, we may simply prove (by considering partial derivatives) that $\boldsymbol{p}_d(u,v)$ is $C^{2n_{i_k}-1}$ continuous on $B_{i_k,j}$, where n_{i_k} is the smoothness degree of the extension function corresponding to D_{i_k}.

When carrying out such complex deformations, the deformation matrices may be constructed independently and then applied in any sequence to the original surface: the product of such deformation matrices is both commutative and associative.

3. Determination of Spine

Using the above approach, a key issue when constructing ribs is how to determine the spine on a parametric surface so that it reflects the desired shape. Users will generally find it more natural to specify a curve on a surface in 3D space instead of a curve in the parametric plane, which furthermore must be in implicit form. Thus, we face two problems. One is how to choose a surface curve as the spine, and the other is how to represent it as an implicit equation in the parametric plane. These problems can be solved by considering them in terms of curve interpolation on a given surface. We base our approach on functional spline methods,[11,12] as follows.

Assuming the surface is regular, we formulate the problem as:

Problem 1. *Given an arbitrary sequence* $\boldsymbol{r}_i, i = 1, \ldots, s$ *of points on a* C^r *surface, where* $r \geq 2$, *find an interpolation curve on the surface passing through them with tangent direction* t_i *at corresponding point* $\boldsymbol{r}_i$.

Suppose $\boldsymbol{r}(u,v) = (x(u,v), y(u,v), z(u,v))^T$, $u, v \in [0,1]$ is a regular surface with continuity C^{r_1}, and $u = u(t), v = v(t)$, $t \in [a,b]$ represents a curve with continuity C^{r_2} in the (u,v) plane. Substituting this curve into above surface equation gives a space curve with continuity $C^{\min[r_1,r_2]}$:

$$\boldsymbol{r}(t) = (x(u(t),v(t)), y(u(t),v(t)), z(u(t),v(t))) = (x^*(t), y^*(t), z^*(t))^T, \quad t \in [a,b]$$

which is contained in the surface.

As the surface is defined by a mapping

$$\boldsymbol{r} : [0,1] \times [0,1] \to R^3,$$

and since we assume the surface is regular, the mapping is one-to-one, so the tangent mapping induced by it is an isomorphic mapping between tangent spaces of the planar domain and the surface at corresponding points. Using $T_r(\boldsymbol{r}(t_0))$ and $T_\alpha(\boldsymbol{\alpha}(t_0))$ to denote the tangent spaces of the surface at the point $\boldsymbol{r}(t_0)$ and the corresponding planar domain at the point $\boldsymbol{\alpha}(t_0)$ respectively, then the tangent mapping

$$d\boldsymbol{r} : T_\alpha(\boldsymbol{\alpha}(t_0)) \to T_r(\boldsymbol{r}(t_0)) \subset R^3$$

is a linear one-to-one mapping (isomorphism). Differential geometry of surfaces[13] tells us that this mapping can be expressed in the matrix form

$$d\boldsymbol{r}(\boldsymbol{X}) = \begin{pmatrix} \partial x/\partial u & \partial x/\partial v \\ \partial y/\partial u & \partial y/\partial v \\ \partial z/\partial u & \partial z/\partial v \end{pmatrix} \boldsymbol{X} = \begin{pmatrix} \boldsymbol{r}_u & \boldsymbol{r}_v \end{pmatrix} \boldsymbol{X}, \tag{3}$$

in (3) satisfies $rank(\boldsymbol{r}_u \quad \boldsymbol{r}_v) = 2$. This allows us to conclude that:

At corresponding points, the tangent vector of a surface curve on a regular surface and the tangent vector of its original image curve uniquely determine each other.

3.1. G^1 *interpolation on surface*

To define the spine, we perform piecewise interpolation between each pair of points $\boldsymbol{r}_1, \boldsymbol{r}_2$; we also assume we have the corresponding tangent vectors $\boldsymbol{T}_1, \boldsymbol{T}_2$. As $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ lie on a regular surface, they correspond to unique *original image points* in the parameter plane which we call $\boldsymbol{\alpha}_1 = (u_1, v_1)$, $\boldsymbol{\alpha}_2 = (u_2, v_2)$ We could find these image points by applying Newton iteration, but we prefer another method[14] that has better stability and quicker convergence. Other approaches based on resultants,[15] implicitization using Gröbner bases,[16] and implicitization using Sederberg's method[17] could also be used to solve this problem.

The previous section tells us that the tangent vectors $\boldsymbol{T}_1$, $\boldsymbol{T}_2$ determine the corresponding original tangent image vectors under the tangent mapping $d\boldsymbol{r}$. We denote the image tangent vectors by $\boldsymbol{t}_1$, $\boldsymbol{t}_2$: these are the tangent vectors at points $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ of the plane curve uniquely determined by the desired interpolation curve on the surface. We note that $\boldsymbol{t}_1, \boldsymbol{t}_2$ can be computed from Eq. (3).

The problem of surface interpolation is thus reduced to interpolation in the parameter plane, which can easily be solved.[11,12] Let $x_1 = (u_1, v_1), x_2 = (u_2, v_2)$ be the coordinate vectors of the points $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ in parametric plane respectively, let $g_i(\boldsymbol{x}) = n_i.(\boldsymbol{x} - x_i) = 0$ be the normal equation of the straight line l_i which passes through α_i in the direction of $\boldsymbol{t}_i, i = 1, 2$, and let $g_{12}(\boldsymbol{x}) = \boldsymbol{n}_{12}.(\boldsymbol{x} - x_1) = 0$ be the equation of the line joining $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$, where $\boldsymbol{x} = (u, v)$. We can always write these equations so they satisfy $g_1(\boldsymbol{x}_2) > 0, g_2(\boldsymbol{x}_1) > 0$, and we do so. Following Li,[12] the desired interpolation curve with tangent vectors $\boldsymbol{t}_1, \boldsymbol{t}_2$ at points $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ is:

$$(1 - \mu)g_1 g_2 + \mu g_{12}^2 = 0, \quad 0 < \mu < 1. \tag{4}$$

The constant μ is a free parameter which can be used to modify the shape of the interpolation curve.

Generalising this to a sequence of points to be interpolated, labelled $i = 1, \ldots, s$, we may write the piecewise solution as $\boldsymbol{r}(u, v) = (x(u, v), y(u, v), z(u, v))^T$ where

$$(1 - \mu_i)g_i g_{i+1} + \mu_i g_{i,i+1}^2 = 0, \quad 0 < \mu_i < 1, \tag{5}$$

where $g_{i,i+1}(\boldsymbol{x}) = \boldsymbol{n}_{i,i+1}.(\boldsymbol{x} - x_i) = 0$ is the normal equation of the straight line between $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_{i+1}$. All $\mu_i, i = 1, \ldots, s-1$ can be chosen independently, allowing interactive modification of the shape of the spine, and hence the rib.

Furthermore differential geometry tells us that the curvature vector, and hence normal curvature and geodesic curvature, of the surface curve determines the curvature of its parametric image curve, and vice versa. This idea could be used in principle to construct a G^2 interpolating curve in the parameter plane if it was required to construct ribs with G^2 continuity.

In practice, although the points to be interpolated on the surface will be given, their tangents will not, and must be estimated. This can readily be done.[11]

3.2. *Bi-arc interpolation on plane*

In sheet metal design, the underlying surface is frequently planar; in such cases it is often desirable that the bounding curves of the deformed region should be offsets of the spine curve. Arc splines are piecewise curves constructed from circular arcs and straight-line segments, and since arc splines have simple offsets, they are useful curves for defining spine curves in such cases. Various methods exist[18–20] for constructing interpolating arc splines with arbitrary shape. We may state the problem to be solved as follows:

Problem 2. *Given an arbitrary sequence* $\boldsymbol{r}_i, i = 1, 2, \ldots, s$ *of points in the plane, and corresponding tangent directions* $\boldsymbol{T}_i$, *find an interpolating curve made of bi-arc curve segments which passes through the points, with direction* $\boldsymbol{T}_i$ *at* $\boldsymbol{r}_i$.

We use Yang's bi-arc interpolation method.[18] Again, we start by considering a single segment of the spline, interpolating two end points $\boldsymbol{r}_1, \boldsymbol{r}_2$ with end unit tangent vectors $\boldsymbol{T}_1, \boldsymbol{T}_2$, and corresponding normals $N_{1,}N_{2,}$. Let ϕ be the directed

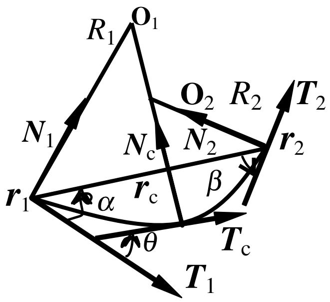

Fig. 2. Bi-arc interpolation curve.

angle between the positive x-axis and tangent $\boldsymbol{T}_1$; counterclockwise angles are considered to be positive in the following. Let $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$ be the centers of the two arc segments to be determined, which have radii R_1 and R_2 respectively. Let $\boldsymbol{r}_c$ be the point where the arcs meet, let the common unit tangent vector there be $\boldsymbol{T}_c$, and the corresponding unit normal vector be N_c. We write the directed angle from $\boldsymbol{T}_1$ to $\boldsymbol{T}_c$ as θ (see Fig. 2). It follows that:

$$N_1 = (-\sin\varphi, \cos\varphi), \quad \boldsymbol{T}_c = (\cos(\varphi+\theta), \sin(\varphi+\theta)), \quad N_c = (-\sin(\varphi+\theta), \cos(\varphi+\theta))$$

Furthermore,

$$\boldsymbol{r}_2 - \boldsymbol{r}_1 - R_1 N_1 = \boldsymbol{r}_2 - \boldsymbol{r}_c - R_1 N_c, \quad \boldsymbol{r}_2 - \boldsymbol{r}_c - R_2 N_c = R_2 N_2.$$

Solving these equations gives:

$$\boldsymbol{r}_1 = \frac{(\boldsymbol{r}_2 - \boldsymbol{r}_1)\cdot(\boldsymbol{T}_2 - \boldsymbol{T}_c)}{(N_1 - N_c)\cdot(\boldsymbol{T}_2 - \boldsymbol{T}_c)}, \quad \boldsymbol{r}_2 = \frac{(\boldsymbol{r}_2 - \boldsymbol{r}_c)\cdot(\boldsymbol{r}_2 - \boldsymbol{r}_c)}{2(\boldsymbol{r}_2 - \boldsymbol{r}_c)\cdot N_c},$$

$$\boldsymbol{O}_1 = \boldsymbol{r}_1 + R_1 N_1, \quad \boldsymbol{O}_2 = \boldsymbol{r}_c + R_2 N_c, \quad P_c = \boldsymbol{r}_1 + R_1(N_1 - N_c)$$

where θ is a free parameter which enables us to interactively modify the shape of the bi-arc curve segment. Let α, β be the two directed angles between the vector $\boldsymbol{r}_2 - \boldsymbol{r}_1$ and $\boldsymbol{T}_1$, and $\boldsymbol{T}_2$, respectively. If α and β have the same sign, we can construct a C-shaped bi-arc curve segment whose optimal shape (in a fairness sense) is obtained when $\theta = \alpha$. Otherwise the bi-arc segment is S-shaped with optimal solution at $\theta = (3\alpha - \beta)/2$.

We now turn to the problem involving a sequence of points $\boldsymbol{r}_i, i = 1, 2, \ldots, s$. Here one can first specify $\boldsymbol{T}_1$ and $\boldsymbol{T}_s$, and then find optimal tangent vectors

Table 1. Computation times for various examples. Number of data points denotes the number of points used to draw the figure with Matlab.

	Fig. 3	Fig. 5	Fig. 8	Fig. 9	Fig. 10	Fig. 11	Fig. 13
CPU time	0.19	0.21	0.07	0.10	0.09	0.09	0.12
Number of data points	10^4	10^4	10^4	10^4	10^4	10^4	10^4

$\boldsymbol{T}_i, i = 2, \ldots, s-1$ corresponding to these intermediate points.[19] Alternatively we may estimate all tangent directions together,[10] before carrying out bi-arc interpolation as above.

3.3. *Main steps of the deformation method*

We now give the whole process for constructing the ribs on the surface, starting with spine construction as discussed above:

(1) Determine the spine using G^1 interpolation on the surface, or bi-arc interpolation in the plane.
(2) Construct the extension functions and deformation matrix.
(3) Multiply the equation of the surface by the deformation matrix.
(4) Interactively adjust the parameters such that the resulting ribs have the desired final shape.

Examples of construct ribs and other structures are shown in Figs. 3–11. As well as being used to construct ribs on surfaces, our method can also be used to generate other deformation effects. See, for example, Figs. 13–14, which are produced by deforming a plane.

Fig. 3. An arbitrary-shaped bead on surface.

Fig. 4. Creating the base of a bowl and two ribs near its brim.

Fig. 5. Two intersecting ring-shaped grooves on a Bézier surface.

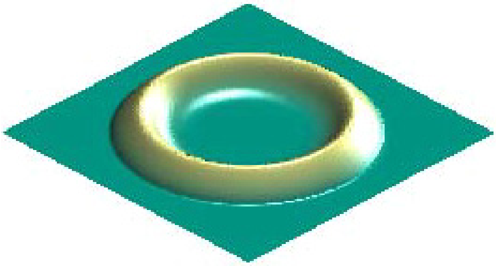

Fig. 6. Ring-shaped rib with circular spine.

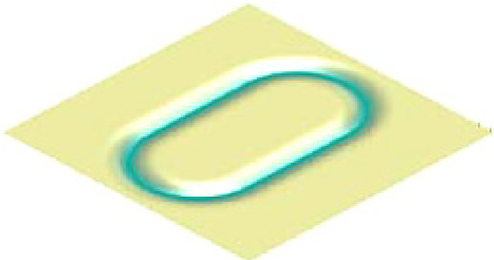

Fig. 7. Closed rib with arc spline spine.

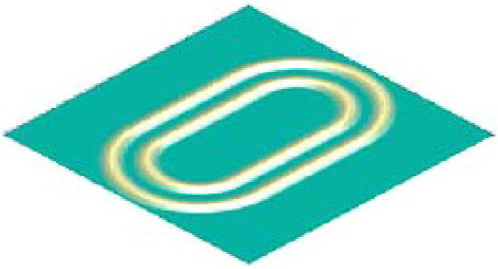

Fig. 8. Rib with repeat count 2 with arc spline spine.

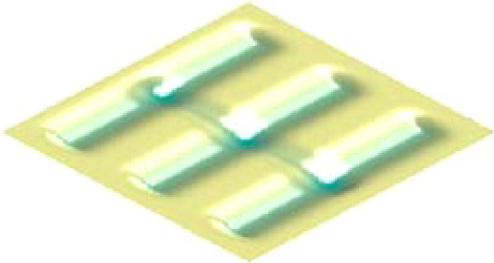

Fig. 9. Intersecting ribs with repeat counts 1 and 3.

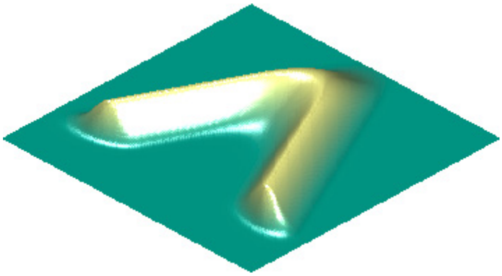

Fig. 10. V-shaped rib with a arc spline spine.

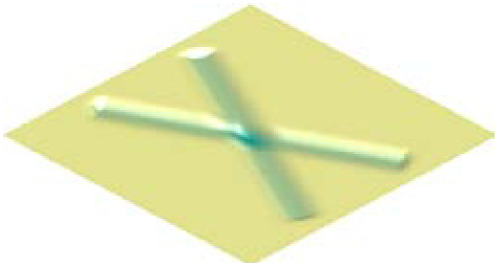

Fig. 11. X-shaped rib.

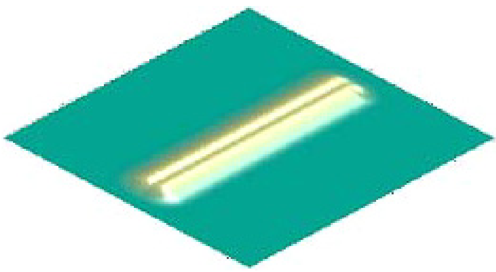

Fig. 12. I-shaped rib with a groove on its top.

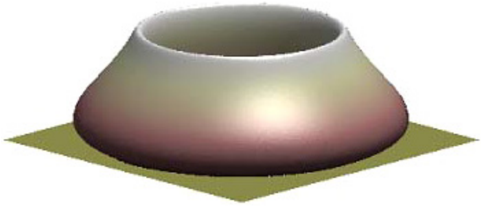

Fig. 13. A bowl generated by deforming a plane.

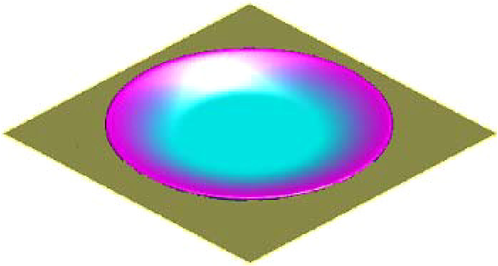

Fig. 14. A dish generated by deforming a plane.

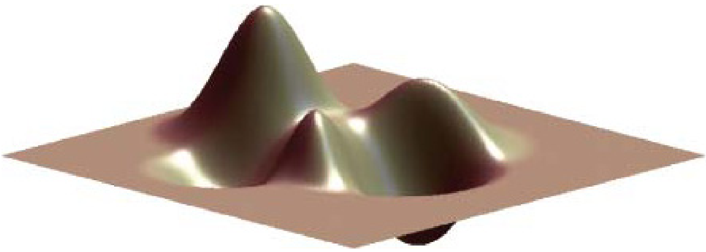

Fig. 15. A piecewise G^2-continuous surface.

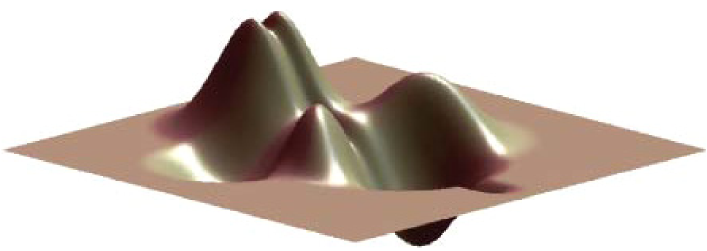

Fig. 16. An I-shaped groove applied to the surface in Fig. 15.

4. Implementation and Examples

We use the following parametric surfaces to demonstrate our method: a paraboloid, a Bézier surface and a plane. The paraboloid S_1 is $z = \frac{1}{8}(-(u-1)^2 - (v-1)^2 + 2) + 2$. The control points of the Bézier surface S_2 (see Fig. 5) are $(-4,-4,2)$, $(0,-5,2.5)$, $(4,-4,2)$, $(-4,0,2.5)$, $(0,1,4.5)$, $(4,0,2.5)$, $(-4,4,2)$, $(0,5,2)$, $(4,4,2)$.

Figure 3 shows a deformation of S_1, where the spine is a free-form curve obtained by the method in Sec. 3.1. Figures 4, 5 show deformations of S_1 and S_2 where the spines are circles. Figures 6–14 show deformations applied to a plane; the spines are circles in Figs. 6, 13, 14, arc splines in Figs. 7, 8, 10, and line segments in Figs. 9, 11, 12. Note that Figs. 4, 5, 9, 11, 12 illustrate application of multiple deformations as given by Eq. (2). Repeated deformation effects are showed in Figs. 4, 8, 9, 13, 14, where the main parameters were chosen as follows: $\omega_1 = \omega_2 = \omega_3 = 2$ in Fig. 4; $h_1 = h_2 = 0, h_3 = 0.01$ and ω_3=2 in Fig. 8; $h_1 = h_2 = 0, h_3 = -0.04$ or -0.02 and $\omega_3 = 3$ or 1 in Fig. 9; $h_1 = h_2 = 0.08$, $h_3 = 0.3$, $\omega_1 = \omega_2 = 2$ and $\omega_3 = 1$ in Fig. 13. Figure 15 shows a piecewise continuous surface, and Fig. 16 shows a deformation of this surface.

5. Conclusions

We have given a simple approach to generate ribs, single or repeated, on surfaces, using spine curves which may be open or closed. The method is based on multiplying the surface's equation with a shape matrix, to provide a shape whose details can be controlled interactively by modifying various parameters. Its advantages are its ability to produce a deformation with a precisely controlled location, and specified continuity at its boundary, giving clear advantages over various earlier methods.[1–10] Furthermore, even a quite complicated deformation leads to a final surface expressed using one uniform equation–even if multiple deformations are used. It has the advantage of having few, independent, control parameters with clear geometric meanings. The spine C reflects the general direction of the rib (deformation), while constant S controls the width of the rib and h its height. The repeat count ω_i allows repeating ribs, while the index n controls the continuity at the rib's boundary. Further control is provided by the constant A which has an influence on the location and shape of the spine.

Our discussion is limited to parametric surfaces throughout the paper. While quadratic surfaces are frequently adopted in computer aided geometric design, we note these are monoid surfaces which can readily be parameterized rationally.[15] Unfortunately, however, our method can not be directly applied to triangular meshes or volume data since the deformation operations rely on modifying the equation of the base surface.

Further research will consider the construction of suitable extension functions, and a suitable user interface for applying this method in practice.

Acknowledgment

The authors owe thanks to anonymous referees for their valuable comments and suggestions that greatly improved the manuscript. Moreover, much thanks should be given especially to Prof. R. Martin for critical reading and smoothing the English of the paper. The work reported in this paper got support from the National Natural Science Foundation of China (Nos. 60673026, 50875130, 50805075 and 50875126).

References

1. L. Piegl, "Modifying the shape of rational B-spline, Part II: surfaces," *Computer-Aided Design* **21**(8), 509–518 (October 1989).
2. W. Ma and P. He, "B-spline surface local updating with unorganized points," *Computer-Aided Design* **30**(11), 853–862 (September 1998).
3. J. Ishida, "The general B-spline interpolation method and its application to the modification of curves and surfaces," *Computer Aided Design* **29**(11), 779–790 (November 1997).
4. P. N. Georgiades and D. P. Greenberg, "Locally manipulating the geometry of curved surfaces," *IEEE Computer Graphics and Applications* **12**(1), 54–64 (January 1992).
5. K. Singh and E. Fiume, "Wires: A geometric deformation technique," *Computer Graphics* **32**(3), 405–414 (August 1998).
6. J. C. Léon and P. Veron, "Semiglobal deformation and correction of free-form surface using a mechanical alternative," *The Visual Computer* **13**(3), 109–126 (May 1997).
7. J. C. Léon and P. Trompette, "A new approach towards freeform surfaces control," *Computer Aided Geometric Design* **12**(4), 395–416 (June 1995).
8. S. M. Hu, Y. F. Li, T. Ju and X. Zhu, "Modifying the shape of NURBS surfaces with geometric constraint," *Computer Aided Design* **33**(12), 903–912 (December 2001).
9. J.-L. Mari and J. Sequeira, "Closed free-form surface geometrical modeling a new approach with global and local characterization," *International Journal of Image and Graphics* **4**(2), 241–262 (April 2004).
10. X. P. Wang, R. R. Zhou, Z. L. Ye and L. Y. Zhang, "Periodic deformation of parametric surface", *Chinese Journal of Mechanical Engineering* (English Edition) **17**(3), 394–398 (September 2004).
11. E. Hartmann, "G^2 interpolation and blending on surfaces," *The Visual Computer* **12**(4), 181–192 (April 1996).
12. J. Li, J. Hoschek and E. Hartmann, "G^{n-1} functional splines for interpolation and approximation of curves, surfaces and solids," *Computer Aided Geometric Design* **7**(2), 209–220 (February 1990).
13. M. P. do Carmo, *Differential Geometry of Curves and Surfaces*, Englewood Cliffs, New Jersey Prentice-Hall, Inc. 1976.
14. S.-M. Hu, J.-G. Sun, T.-G. Jin and G.-Z. Wang, "Computing the parameters of points on NURBS curves and surfaces via moving affine frame method," *Journal of Software* (in Chinese) **11**(1), 49–53 (January 2000).
15. T. W. Sederberg, "Algebraic geometry for computer aided geometric design," *IEEE Computer Graphics and Applications* **6**(1), 52–59 (January 1986).
16. C. M. Hoffmann, "Implicit curves and surfaces in CAGD," *IEEE Computer Graphics and Applications* **13**(1), 79–88 (January 1993).
17. T. W. Sederberg, D. C. Anderson and R. N. Goldman, "Implicit representation of parametric curves and surfaces," *Computer Vision, Graphics and Image Processing* **28**(1), 72–84 (January 1984).
18. X. Yang, "Efficient circle arc interpolation based on active tolerance control," *Computer-aided Design* **34**(13), 1037–1046 (November 2002).

19. B. Q. Su and D. Y. Liu, *Computational Geometry — Curve and Surface Modeling*, Academic Press, San Diego (1989).
20. G. J. Wang, G. Z. Wang and J. M. Zheng, *Computer Aided Geometric Design*, China: Higher Education Press, Berlin: Spring-Verlag (2001).

Xiaoping Wang received his MS degree in Pure Mathematics from Zhengzhou University, China, 1995, and his PhD degree in Aerospace Manufacture Engineering from Northwestern Polytechnical University, China, in 2002. From July 2002 to July 2004, he worked as a postdoctoral research fellow at Research Center of CAD/CAM Engineering at Nanjing University of Aeronautics and Astronautics (NUAA). Now he is an associate professor at the Department of Aircraft Manufacturing Engineering, NUAA, China. His current research interests cover CAD/CAM, computer aided geometric design, computer graphics, etc.

Shenglan Liu obtained her BS, MS and PhD degrees from NUAA, China, in 1994, 1997 and 2004, respectively, all in Aerospace Manufacture Engineering. She is currently a research fellow/associate in Computer Science at Cardiff University, UK. Her research interests include inverse engineering, CAD/CAM, computer graphics and so on.

Liyan Zhang received her BS degree in Aircraft Design from NUAA, China, in 1988, her MS degree in Engineering Mechanics from Xi'an Jiaotong University, China, 1991, and her PhD in Aerospace Manufacture Engineering from NUAA, China, in 2001. From November 2003 to November 2004, she was a visiting professor with the department of Computer science, Arizona State University, USA. She is now a professor in the research center of CAD/CAM Engineering, NUAA. She is mainly devoted to the research of digitized remodeling based on measured data, computer graphics and so on.