Xu G, Wang GZ, Chen XD. Free-form deformation with rational DMS-spline volumes. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 23(5): 862–873 Sept. 2008

Free-Form Deformation with Rational DMS-Spline Volumes*

Gang Xu[1,2] (徐 岗), Guo-Zhao Wang[1,2] (汪国昭), and Xiao-Diao Chen[3,4,*] (陈小雕)

[1] *Institute of Computer Graphics and Image Processing, Zhejiang University, Hangzhou 310027, China*

[2] *Department of Mathematics, Zhejiang University, Hangzhou 310027, China*

[3] *College of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China*

[4] *INRIA, Nancy, France*

E-mail: xugangzju@yahoo.com.cn; wanggz@zju.edu.cn; xiaodiao@hdu.edu.cn

Received November 15, 2007; revised May 6, 2008.

Abstract In this paper, we propose a novel free-form deformation (FFD) technique, RDMS-FFD (Rational DMS-FFD), based on rational DMS-spline volumes. RDMS-FFD inherits some good properties of rational DMS-spline volumes and combines more deformation techniques than previous FFD methods in a consistent framework, such as local deformation, control lattice of arbitrary topology, smooth deformation, multiresolution deformation and direct manipulation of deformation. We first introduce the rational DMS-spline volume by directly generalizing the previous results related to DMS-splines. How to generate a tetrahedral domain that approximates the shape of the object to be deformed is also introduced in this paper. Unlike the traditional FFD techniques, we manipulate the vertices of the tetrahedral domain to achieve deformation results. Our system demonstrates that RDMS-FFD is powerful and intuitive in geometric modeling.

Keywords free-form deformation, rational DMS-spline volume, control lattice of arbitrary topology, multiresolution deformation, direct manipulation

1 Introduction

1.1 Motivation and Contribution

The design of complex models is a key problem in geometric modeling[1−3]. One of the most powerful tools for deforming a complex model is free-form deformation (FFD), which was first introduced by Sederberg and Parry[4]. By using this method, the users first generate some control points called a lattice to contain a 3D model to be deformed, and then modify the control points to deform the parametric space inside. Finally, the deformed parametric space is mapped onto a target model automatically. A lot of related approaches are based on this method, and it has been integrated into most of leading commercial computer animation software systems as their modeling and animation components, for example, Maya, Softimage XSI, 3DS MAX etc. However, each method of free-form deformation has its own advantages and disadvantages. We will discuss them in next section. In [5], Bechmann and Gain proposed some criteria to assess the FFD technique, such as versatility, ease of use, correctness and efficiency. In particular, local property, multiresolution and direct manipulation are also very important for free-form deformation. Unfortunately, few of the existing methods satisfy user's demands in all the aspects listed above.

In this paper, we propose a novel free-form deformation technique based on rational DMS-spline volume, which satisfies most of the criteria listed above. We call this method RDMS-FFD. DMS-splines were proposed by Dahmen *et al.* in [1]. We generalize the results presented in [6, 7] and give the definition of rational DMS-spline volumes, which generalizes DMS-spline surfaces with planar triangulation domains to tetrahedral domains, with arbitrary topology. The rational DMS-spline volumes have many beautiful properties, such as convex hull property, local support, continuity and affine invariance. More importantly, they can be defined by the control lattice of arbitrary topology. Our technique inherits these properties and provides various important advantages, which we list in the following summary of the main contributions of our work.

- Unlike the traditional FFD methods, we

Regular Paper

This work is supported by the National Natural Science Foundation of China under Grant Nos. 60773179 and 60473130, and the National Basic Research 973 Program of China under Grant No. 2004CB318000.

*Corresponding Author

manipulate the vertex of the tetrahedral domain to achieve deformation results. The control lattices will be updated according to the new tetrahedral domain. In fact, the control lattices have the same shape as the tetrahedral domain, only some control points will be located at the edges of the tetrahedral domain. Hence, ease of use is improved.

• The tetrahedral domain can be arbitrary topology and approximate any complex shapes tightly and automatically. Hence, we can easily predict the deformation results from the manipulating of vertices of the tetrahedral domain.

• DMS-spline has smooth basis functions defined on arbitrary topology. Hence, the smoothness of the deformation space can be guaranteed. This solves the problem presented in [8].

• The deformation method has local and multiresolution property. Hence, the deformation technique has elaborate sculpture ability.

• Direct manipulation of deformation can be achieved by combining weights change and lattices manipulation. We will discuss it in detail in Subsection 4.4.

1.2 Overview

The rest of the paper is organized as follows. In Section 2, we introduce some related work on FFD techniques and DMS-splines. Section 3 briefly reviews DMS-spline volumes[7] and presents rational DMS-spline volumes. Section 4 describes the deformation algorithm in detail which includes the generation of the tetrahedral domain and control lattice, the attachment of the object to the rational DMS-spline volume, and multiresolution deformation technique. Section 5 presents some experimental results and discusses the performance. Conclusions and future work are presented in Section 6.

2 Related Work

In this section we present an overview of a number of existing free-form deformation techniques and some related work on DMS-splines.

Free-Form Deformation. Free-form deformation technique, firstly proposed by Sederberg and Parry[4], is widely available and almost all the subsequent methods are based on it. Griessmair and Purgathofer proposed a new FFD method based on B-Spline volumes, and optimized the mesh division after deformation[9]. Kalra *et al.* proposed a Rational Free-Form Deformation (RFFD) method which uses a rational parametric volume to simulate the movement of facial muscles[10]. The method proposed by Lamousin and Waggenspack was based on NURBS volumes and succeeded in raising the flexibility of FFD[11]. Raviv and Elber developed a three dimensional interactive sculpting paradigm that employs a collection of scalar uniform trivariate B-spline functions[12]. Kagan and Fischer presented a CAE system using the deformation method based on finite element B-splines[13]. However, the control lattice of the above methods should be a regular parallelepiped or a uniformly arranged shape. Hence, the users cannot predict the deformation results correctly from the manipulation of the control lattice.

To overcome this limitation, many FFD techniques that use control lattices with arbitrary topology have been proposed in recent years. Coquillart proposed an extension of free-form deformation (EFFD), which uses several low resolution lattices, called "chunk", for deformation[14]. Coquillart and Jancéne used EFFD method to build animations called Animated Free-Form Deformation (AFFD)[15]. But it has a continuity problem. By using the Catmull-Clark subdivision scheme, MacCracken and Joy further extended the capability of FFD by introducing lattices of arbitrary topology[16]. However, the smoothness of the deformation space is not guaranteed. Furthermore, how to generate the control lattice is also difficult. Bechmann *et al.* proposed continuous FFD (CFFD) based on barycentric coordinates and Bézier tetrahedrons[17]. Its control lattice is also arbitrary topology, but keeping the deformation continuous between tetrahedrons requires defining constraints on the displacement of control points. Thus the manipulation of the control lattice is no longer free-form. Moccozet and Magnenat-Thalmann proposed Dirichlet FFD approach based on the Voronoi structure defined within the convex hull of a set of points[18]. Though there is no restriction on the shape of the volume, it is hard to manipulate the influence of a single control point. By employing weighted T-spline volume as deformation tools, Song and Yang proposed a more flexible and adaptive FFD method[19]. But for some special cases, the control lattice cannot approximate the shape of object very well. Ju *et al.* generalized mean-value coordinates from closed 2D polygons to closed triangular meshes and applied them to free-form deformation[20]. But it is only appropriate for global deformation. Feng *et al.* proposed a deformation technique with subdivision surface of arbitrary topology[8]. However, it is hard to achieve direct manipulation of deformation. In [21], Yoon and Kim proposed a sweep-based approach to the free-form deformation of 3D objects. Unlike the previous methods, they approximate deformable parts using sweep surfaces. Hierarchy-based interaction of deformation and volume preserving

property are its main advantages, but multiresolution deformation is difficult to be achieved.

Based on this background of FFD methods with control tools of arbitrary topology, we propose RDMS-FFD by combining more deformation techniques than previous FFDs in a consistent framework. The details will be given in the following sections.

There are many other FFDs based on surfaces or curves, but most of them are only suitable for global deformation. Feng *et al.* provided an FFD method by using two parametric surfaces called shape surface and height surface[22]. Feng and Peng proposed an accurate B-Spline FFD method[23]. Kazuya *et al.* proposed a similar method which adopted a triangular mesh as the deformation control tool[24]. Hua and Qin presented a modified FFD in which they employ a scalar field as the embedding space instead of a volume[25]. Lazarus *et al.* used an axis, instead of using a lattice, to provide an efficient and intuitive deformation method called Axial Deformations (AxDf)[26]. Chang and Rockwood used a Bézier curve to define the desirable skeleton of the deformed object[27]. Singh and Fiume used wires for deformation[28].

In order to improve the controllability of FFD techniques, several constraint FFD methods have been proposed. To overcome the difficulty in getting the deformed object to pass through desired points precisely, the direct manipulation FFD methods were proposed. Hsu *et al.*[29] adopted a least-square fitting approach to determine the movement of the control points. Hu *et al.*[30] solved this problem explicitly using constraint optimization. The direct manipulation of a generalized cylinder was presented in [31]. Aubert and Bechmann proposed a volume-preserving FFD approach[32]. The algorithm presented in [33] is another kind of constrained FFD, where the developability of a patch undergoing FFD is preserved.

DMS-Splines. Dahmen *et al.* proposed DMS-splines from the point of view of blossoming[1]. Fong and Seidel presented several useful properties of DMS-splines, such as affine invariance, convex hull, locality, and smoothness[34]. Greiner and Seidel presented some applications of multivariate B-spline algorithms in computer graphics and geometric design[35]. Pfeifle and Seidel proposed a faster evaluation technique for quadratic bivariate DMS-spline surfaces[36]. Franssen *et al.* proposed an efficient evaluation algorithm for DMS-spline surfaces of arbitrary degree[37]. Qin and Terzopoulos presented rational DMS-spline surfaces and their dynamic generalization[6]. He and Qin presented a surface reconstruction approach by using DMS-splines surface[38]. Hua *et al.* introduce trivariate DMS-spline volume and employed it in solid representation and modeling[7]. He *et al.* presented rational spherical DMS-spline for genus zero shape modeling[39]. He *et al.* proposed an automatic and efficient method of generating visually pleasing, high-quality DMS-spline surfaces of arbitrary topology for shape fairing[40]. Gu *et al.* introduced manifold DMS-splines, which generalize DMS-spline surfaces with planar domains to manifold domains with arbitrary topology with or without boundaries[41].

3 DMS-Splines and Rational DMS-Spline Volumes

DMS-splines, introduced by Dahmen, Micchelli and Seidel in [1], are based on the simplex splines. Furthermore, DMS-spline is a multivariate B-spline scheme, and its univariate (surface) scheme is also called triangular B-spline. Throughout this paper, we employ the trivariate (volume) scheme to deform the free-form objects. Now we will firstly review the formulation of the trivariate simplex splines presented in [7].

A trivariate simplex spline $M(\boldsymbol{x}|\boldsymbol{x}_0,\ldots,\boldsymbol{x}_{n+3})$ of degree n is a function of $\boldsymbol{x}\in\mathbb{R}^3$ over the half open convex hull of a point set $\boldsymbol{V}=[\boldsymbol{x}_0,\ldots,\boldsymbol{x}_{n+3})$, depending on the $n+4$ knots $\boldsymbol{x}_i\in\mathbb{R}^3, i=0,1,\ldots,n+3$. The basis function of trivariate simplex splines can be defined recursively as follows. When $n=0$,

$$M(\boldsymbol{x}|\boldsymbol{x}_0,\ldots,\boldsymbol{x}_{n+3})=\begin{cases}\dfrac{1}{6|\mathrm{Vol}_{\mathbb{R}^3}(\boldsymbol{x}_0,\ldots,\boldsymbol{x}_3)|}, & \boldsymbol{x}\in[\boldsymbol{x}_0,\ldots,\boldsymbol{x}_3),\\ 0, & \text{otherwise.}\end{cases}$$

When $n>0$, select four points $\boldsymbol{W}=[\boldsymbol{x}_{k_0},\boldsymbol{x}_{k_1},\boldsymbol{x}_{k_2},\boldsymbol{x}_{k_3}]$ from $\boldsymbol{V}$, such that $\boldsymbol{W}$ is affine independent, then

$$M(\boldsymbol{x}|\boldsymbol{x}_0,\ldots,\boldsymbol{x}_{n+3})=\sum_{j=0}^{3}\lambda_j(\boldsymbol{x}|\boldsymbol{W})M(\boldsymbol{x}|\boldsymbol{V}\backslash\{\boldsymbol{x}_{k_j}\}),$$

where $\sum_{j=0}^{3}\lambda_j(\boldsymbol{x}|\boldsymbol{W})=1$ and $\sum_{j=0}^{3}\lambda_j(\boldsymbol{x}|\boldsymbol{W})\boldsymbol{x}_{k_j}=\boldsymbol{x}$. In fact, $\lambda_j(\boldsymbol{x}|\boldsymbol{W})$ are the barycentric coordinates of $\boldsymbol{x}$ with respect to $\boldsymbol{W}$.

In the following, we will review the construction of DMS-spline volume. Let Ω be an arbitrary proper tetrahedralization of $\mathbb{R}^3$ or some bounded domain $D\subset\mathbb{R}^3$. The proper tetrahedralization means that every pair of tetrahedral domain are disjoint, or share exactly one vertex, one edge, or one face. Next, with every vertex $\boldsymbol{x}$ of Ω, we associate a cloud of knots $[\boldsymbol{t}_0,\boldsymbol{t}_1,\ldots,\boldsymbol{t}_n]$, where $\boldsymbol{t}_0=\boldsymbol{t}$. For every tetrahedron $\boldsymbol{I}=(\boldsymbol{p},\boldsymbol{q},\boldsymbol{r},\boldsymbol{s})$, we require

• all the tetrahedron $[\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l]$ with $i+j+k+l \leqslant n$ are non-degenerate;

• the set

$$Z = \text{interior}(\bigcap_{i+j+k+l \leqslant n} [\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l])$$

satisfies

$$Z \neq \emptyset;$$

• if $\boldsymbol{I}$ has a boundary triangle, the knots associated with the boundary triangle must lie outside of Ω.

Then the trivariate DMS-spline basis function $N_\beta^{\boldsymbol{I}}(\boldsymbol{u})$ is defined by means of trivariate simplex spline $M(\boldsymbol{u}|\boldsymbol{V}_\beta^{\boldsymbol{I}})$ as

$$N_\beta^{\boldsymbol{I}}(\boldsymbol{u}) = |d(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)| M(\boldsymbol{u}|\boldsymbol{V}_\beta^{\boldsymbol{I}}),$$

where β is the 4-tuple (i, j, k, l),

$$\boldsymbol{V}_\beta^{\boldsymbol{I}} = [\boldsymbol{p}_0, \ldots, \boldsymbol{p}_i, \boldsymbol{q}_0, \ldots, \boldsymbol{q}_j, \boldsymbol{r}_0, \ldots, \boldsymbol{r}_k, \boldsymbol{s}_0, \ldots, \boldsymbol{s}_l].$$

$d(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)$ is six times of the volume of $(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)$.

A degree n DMS-spline volume $\boldsymbol{S}(\boldsymbol{u})$ over Ω is then defined as

$$\boldsymbol{S}(\boldsymbol{u}) = \sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n} \boldsymbol{c}_\beta^I N_\beta^I(\boldsymbol{u}), \tag{1}$$

where $\boldsymbol{c}_\beta^{\boldsymbol{I}} \in \mathbb{R}^3$ are the control points.

Generalizing (1) by associating a weight ω_β^I with each control point, we define rational DMS-spline volume as the combination of a set of piecewise rational functions:

$$\boldsymbol{F}(\boldsymbol{u}) = \frac{\boldsymbol{P}(\boldsymbol{u})}{\boldsymbol{Q}(\boldsymbol{u})} = \frac{\sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n} \omega_\beta^{\boldsymbol{I}} \boldsymbol{c}_\beta^{\boldsymbol{I}} N_\beta^{\boldsymbol{I}}(\boldsymbol{u})}{\sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n} \omega_\beta^{\boldsymbol{I}} N_\beta^{\boldsymbol{I}}(\boldsymbol{u})}.$$

Note that when we set the weights of each control point to be 1, the rational DMS-spline volume is just the DMS-spline volume. The rational DMS-spline volumes can be also considered as the trivariate generalization of the triangular NURBS presented in [6]. They have many properties of the non-rational schemes, such as convex hull property, local support and affine invariance. Moreover, they have some additional properties:

• like the DMS-spline volumes, rational DMS-spline volumes and their rational basis functions are also C^{n-1} continuous if the knots are in general position, where n is the degree of rational DMS-spline;

• the weights of rational DMS-spline volumes are extra degrees of freedom which influence local shape. If a weight is increased, the volume will move towards the corresponding control point.

The proof of these properties is similar to the proof in [6, 39]. In order to improve the performance of the evaluation algorithm for rational DMS-spline volume, in this paper, we consider a more restricted case by sharing respective control points and weights along common triangles of two adjacent tetrahedrons in tetrahedral domain.

For rational DMS-spline volume with shared control points/weights, we can prove[38]

$$\boldsymbol{P}(\boldsymbol{u}) = \sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n-1} \hat{\boldsymbol{c}}_\beta^{\boldsymbol{I}}(\boldsymbol{u}) N_\beta^{\boldsymbol{I}}(\boldsymbol{u}), \tag{2}$$

$$\boldsymbol{Q}(\boldsymbol{u}) = \sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n-1} \hat{\omega}_\beta^{\boldsymbol{I}}(\boldsymbol{u}) N_\beta^{\boldsymbol{I}}(\boldsymbol{u}), \tag{3}$$

where

$$\hat{\boldsymbol{c}}_\beta^I(\boldsymbol{u}) = \sum_{j=0}^{3} \omega_{\beta+e^j}^{\boldsymbol{I}} \boldsymbol{c}_{\beta+e^j}^{\boldsymbol{I}} \lambda_j(\boldsymbol{u}|[\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l]), \tag{4}$$

$$\hat{\omega}_\beta^{\boldsymbol{I}}(\boldsymbol{u}) = \sum_{j=0}^{3} \omega_{\beta+e^j}^I \lambda_j(\boldsymbol{u}|[(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)]), \tag{5}$$

and $e^j = (\delta_{i,j})_{i=0}^3, j = 0, 1, 2$ as the coordinates vectors.

From (2)∼(5), we know that a degree n rational DMS-spline volume $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{P}(\boldsymbol{u})/\boldsymbol{Q}(\boldsymbol{u})$ can be evaluated with the efficiency of a degree $n-1$ volume. In practice, this property is very useful in improving the performance of the algorithm for rendering a rational DMS-spline volume.

Analogously, the directional derivative of $\boldsymbol{F}(\boldsymbol{u})$ with respect to a vector $\boldsymbol{v}$ can be calculated as follows:

$$D_{\boldsymbol{v}} \boldsymbol{F}(\boldsymbol{u}) = \frac{D_{\boldsymbol{v}} \boldsymbol{P}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{u}) D_{\boldsymbol{v}} \boldsymbol{Q}(\boldsymbol{u})}{\boldsymbol{Q}(\boldsymbol{u})}, \tag{6}$$

where

$$D_{\boldsymbol{v}} \boldsymbol{P}(\boldsymbol{u}) = n \sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n-1} \check{\boldsymbol{c}}_\beta^{\boldsymbol{I}}(\boldsymbol{v}) N_\beta^{\boldsymbol{I}}(\boldsymbol{u}), \tag{7}$$

$$D_{\boldsymbol{v}} \boldsymbol{Q}(\boldsymbol{u}) = n \sum_{\boldsymbol{I} \in \Omega} \sum_{|\beta|=n-1} \check{\boldsymbol{\omega}}_\beta^I(\boldsymbol{v}) N_\beta^{\boldsymbol{I}}(\boldsymbol{u}), \tag{8}$$

and

$$\check{\boldsymbol{c}}_\beta^{\boldsymbol{I}}(\boldsymbol{v}) = \sum_{j=0}^{3} \boldsymbol{c}_{\beta+e^j}^{\boldsymbol{I}} \lambda_j(\boldsymbol{v}|[(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)]),$$

$$\check{\boldsymbol{\omega}}_\beta^{\boldsymbol{I}}(\boldsymbol{v}) = \sum_{j=0}^{3} \omega_{\beta+e^j}^{\boldsymbol{I}} \lambda_j(\boldsymbol{v}|[(\boldsymbol{p}_i, \boldsymbol{q}_j, \boldsymbol{r}_k, \boldsymbol{s}_l)]).$$

Note that $\boldsymbol{F}(\boldsymbol{u})$ and $D_{\boldsymbol{v}} \boldsymbol{F}(\boldsymbol{u})$ have the same basis functions. Thus, the value $\boldsymbol{F}(\boldsymbol{u})$ and the first order derivatives can be evaluated simultaneously.

In this paper, we modify and extend the evaluation algorithm on planar triangular B-splines presented in [37] to the rational DMS-spline volume.

From the analytic and geometric properties of rational DMS-spline volume listed above, we know that it is suitable for free-form deformations. In this paper, we mainly employ cubic rational DMS-spline volumes as deformation tools. Cubic rational DMS-spline volumes keep C^2 continuous in the whole parametric volumes. Therefore, there is no continuity problem for deformation.

4 Free-Form Deformation via Rational DMS-Spline Volumes

4.1 Construction of Parametric Domains and DMS-Lattices

As presented in [7], if we locate control points at the edges of the tetrahedral domain of rational DMS-spline volume, the shape and the features represented by the original domain can be well preserved. The difference between the shape of the domain and the resulting rational DMS-spline volume is very small. Hence, in this paper, we first construct the tetrahedral domain that approximates the shape of the object to be deformed. Then the DMS-lattice is constructed according to the following *DMS-lattice update rules*: we place control points on all the vertices of the tetrahedral domain; in addition, we place two control points on each edge of the tetrahedral domain, which divide each edge into three equal-length line segments. Unlike the traditional FFD methods, we employ the tetrahedral domain as deformation tool, that is, we manipulate the vertices of the tetrahedral domain to achieve deformation results. Then, the DMS-lattice will be updated according to the new tetrahedral domain and *DMS-lattice update rules*. The shape of the tetrahedral domain approximates the object shape well, so the deformation control will be intuitive and convenient. The following problem is how to generate the tetrahedral domain that approximates the shape of the object to be deformed tightly.

However, it is difficult to generate the tetrahedral domain according to the object shape automatically since the object representations are different. For a mesh object, we propose a modified version of the methods presented in [42, 43]. For the other object representations, we adopt the method presented in [8]. However, as mentioned in [8], how to generate the DMS-lattices for the non-manifold object robustly and automatically remains an open problem. In the following, we will introduce the generation method of the tetrahedral domain for mesh object. The algorithm consists of two steps (see Fig.1) as follows.

• Construct the closed coarse control mesh M of the original mesh by using the method presented in [44]. The method is based on the progressive convex hull construction algorithm in [42]. For details, see [42, 44].

• Build an inside volumetric graph G_{in} of M by using the method presented in [43]. In order to reduce the vertices of the tetrahedral domain, we set the grid interval grid of the BBC lattice to twice the average edge length of the coarse mesh. For more information, see [43].

After the tetrahedral domain is constructed, the DMS-lattice will be built according to the *DMS-lattice update rules*. The initial weights of control points belonging to the coarse control mesh are set much larger than the other ones in the inside of the tetrahedral domain. The ration of them is 50:1. Hence, we can obtain the desired deformation results by manipulating the control points only belonging to the coarse mesh. Fig.2 presents the tetrahedral domain of the horse model and the corresponding deformation result.

4.2 Parametrization

After the tetrahedral domain and DMS-lattice are constructed, the object will be attached to the rational DMS-spline volume like the traditional FFDs. That is, we should calculate the parametric coordinates for each point of the object to be deformed. Let $\boldsymbol{G} = (x, y, z)$ be an arbitrary point of the original object, RDMS-FFD parameterizes the point $\boldsymbol{G}$ by the DMS-lattice which includes $\boldsymbol{G}$. With the parameters calculated, the point $\boldsymbol{G}$ will be mapped to a new point according to the modified lattice or the modified weights. When the object is embedded into the lattice, we calculate the parametric coordinates of the point $\boldsymbol{G}$ by the following equation

$$\boldsymbol{G} = \frac{\boldsymbol{P}(\boldsymbol{u})}{\boldsymbol{Q}(\boldsymbol{u})} = \frac{\sum_{\boldsymbol{I}\in\Omega}\sum_{|\beta|=n}\omega_\beta^{\boldsymbol{I}}\boldsymbol{c}_\beta^{\boldsymbol{I}}N_\beta^{\boldsymbol{I}}(\boldsymbol{u})}{\sum_{\boldsymbol{I}\in\Omega}\sum_{|\beta|=n}\omega_\beta^{\boldsymbol{I}}N_\beta^{\boldsymbol{I}}(\boldsymbol{u})}, \quad \boldsymbol{u}\in\Omega. \tag{9}$$

Obviously, (9) is a nonlinear equation and the solution is unique.

We will solve (9) by using the method presented in [19]. It has two steps:

• rewrite (13) as an equivalent scalar function $H(\boldsymbol{u}) = \|\frac{\boldsymbol{P}(\boldsymbol{u})}{\boldsymbol{Q}(\boldsymbol{u})} - \boldsymbol{G}\|^2 = 0$;

• solve this equation by using nonlinear conjugate gradient method[45]. The initial solution $\boldsymbol{u}_0 = (u_0, v_0, w_0)$ is set to the point $\boldsymbol{G}$ itself and the decent direction is $-\nabla\boldsymbol{H}_0$, where $\nabla\boldsymbol{H}_0 = (\boldsymbol{F}_u((u_0, v_0, w_0), \boldsymbol{F}_v((u_0, v_0, w_0), \boldsymbol{F}_w((u_0, v_0, w_0))^\top$. $-\nabla\boldsymbol{H}_0$ can be obtained from (6)∼(8).

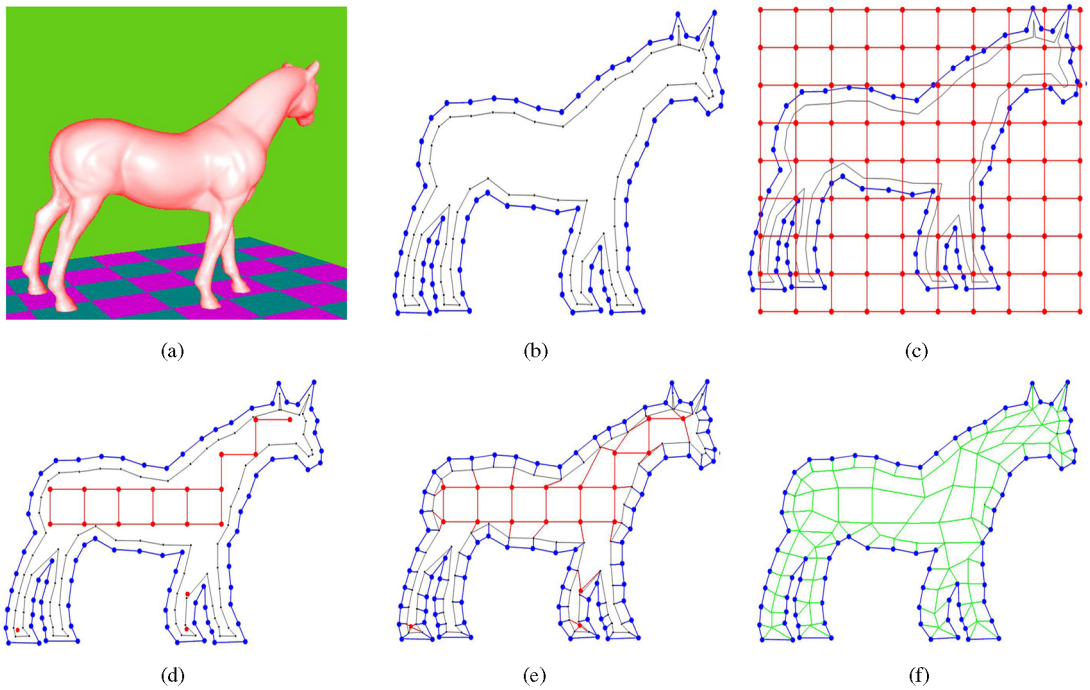

Fig.1. Construction of the tetrahedral domain. (a) Original horse model. (b) Construct an inner shell M_{in} for the coarse mesh M obtained using the method presented in [44]. (c) Embed M_{in} and M in a body-centered cubic (BCC) lattice. (d) Remove lattice nodes outside M_{in}. (e) Build edge connections among M_{in}, M and lattice nodes. (f) Simplification and smoothness.

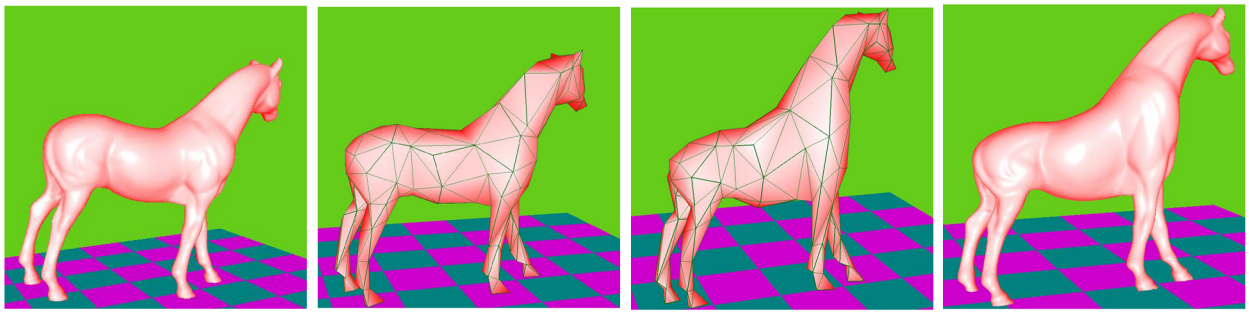

Fig.2. Deformation of the horse model. (a) Initial horse model. (b) Initial tetrahedral domain. (c) Deformed tetrahedral domain. (d) Deformed horse model by (c).

Table 1. Model Size and Parametrization Time

Model	♯ NVOM	♯ NVTD	♯ NCPOS	PT(m)
Horse	42 568	356	463	27.08
Bunny Head	741	182	237	1.59
Chetah	21 478	203	258	13.88
Dolphin	2 217	228	279	5.46
Dinnosaur	23 976	212	324	15.79
Bird	34 126	302	491	22.42

Note: ♯ NVOM: the number of vertices of the original model; ♯ NVTD: the number of the vertices of the tetrahedral domain; ♯ NCPOS: the number of control points generated by octree subdivision for w-TFFD, here the octree depth equals 3; PT: the parametrization time.

Currently, we cannot prove the convergence of iterative algorithm theoretically. If the solution does not converge with limited steps, we can reset a new initial value and solve the equation again. Because rational DMS-spline volumes are arbitrary topology, the recomputation is necessary. In fact, the re-initiation is few in our examples. There are only three times at most. The computational and storage cost of the proposed method is $O(n^3)$. The parameterization runtime statistics of all the examples that we have presented are listed in Table 1. It should be noted that the embedding step is computed only once for the given control lattices. Thus its computational time will not influence the subsequent

deformation interaction.

4.3 Multiresolution Deformation

Multiresolution deformation enables stable reconstruction of the detail information. The deformation technology presented above has a local property inherited from rational DMS-spline volume. However, the local influence region cannot be arbitrarily small for the given lattice, because once the lattice and the tetrahedral domain is given, the influence region of moving a control point is fully determined. To achieve a more elaborate sculpture ability, we propose a multiresolution deformation method.

As we known, the subdivision technique has a multiresolution property. In this paper, we achieve multiresolution deformation by employing the smooth subdivision of tetrahedral mesh presented in [46]. Unlike the method presented in [8], we firstly subdivide the tetrahedral domain with the subdivision scheme in [46], and then we obtain the refined DMS-lattice according to the refined tetrahedral domain and the *DMS-lattice update rules*. The refined tetrahedral domain will have fine resolution while its approximate shape will be similar to the original object. If the refined domain tetrahedral and refined DMS-lattice are adopted as new domain tetrahedral and DMS-lattice and the object is reattached to the rational DMS-spline volume generated from the refined tetrahedral and DMS-lattice, the deformation will have a better local property than the original tetrahedral domain. In our implementation, we firstly deform the object by the raw tetrahedral domain (see Figs.3(b) and 3(c)). Then the refined tetrahedral domain is used as the new tetrahedral domain (see Fig.3(d)). Finally, we reattach the globally deformed object on the new rational DMS-spline volume to obtain fine local deformation (see Figs.3(e) and (f)). This interactive process continues until we obtain a satisfying deformation result according to the user's intention.

4.4 Deformation Algorithm

The main steps for the RDMS-FFD algorithm are as follows.

• Construct the tetrahedral domain and DMS-lattice. Set initial weights for rational DMS-spline volume.

• Calculate the parametric coordinates (u, v, w) for each point on the object to be deformed.

• Manipulate the vertices of the tetrahedral domain or weights of the control points and update the DMS-lattice, evaluate the new locations of the points according to the new tetrahedral domain and DMS-lattice. We will obtain a raw global deformation.

• Use multiresolution deformation technique to obtain satisfying results.

RDMS-FFD supports direct manipulation. For a given objective point, we only move some vertices of the tetrahedral domain towards the objective point, and increase the weight of the corresponding control point that is closest to the objective point, then all affected points with the rational DMS-volume will move towards the objective point (see Fig.4). Fig.5 shows an example with the same control mesh but the different weights at the control points with yellow color.

5 Implemented Results and Discussion

In this section, we will present some experimental results achieved with our deformation system. Our deformation system has functions such as construction of the closed coarse control mesh of the original mesh, generation of the inside volumetric graph, modeling of rational DMS-spline volumes, direct deformation, multiresolution deformation etc.

Fig.2(b) illustrates the tetrahedral domain for the horse model shown in Fig.2(a). The modified tetrahedral domain is shown in Fig.2(c) and the corresponding deformation result is presented in Fig.2(d).

Fig.3 is an example of multiresolution deformation control for a bunny head. Fig.3(a) presents the original model. The modified raw tetrahedral domain and corresponding deformed model are shown in Figs.3(b) and 3(c). The refined tetrahedral domain is illustrated in Fig.3(d). Figs.3(e) and 3(f) show the deformed fine tetrahedral domain and the corresponding deformation result.

Fig.4 shows several examples of the direct manipulation of deformation of the bunny's ear. Fig.4(a) presents the original model and two objective points. Fig.4(b) shows the direct deformation results. Other two examples are shown in Figs.4(c) and 4(d). Fig.5 illustrates the effects of the weights.

Fig.6 shows some deformation results of the dolphin model. Fig.6(a) illustrate the original model. Figs.6(b), 6(d), 6(f) and 6(h) show the deformed tetrahedral domain. Figs.6(c), 6(e), 6(g) and 6(i) show the corresponding deformation results.

Fig.7 shows the deformation results of dinosaur model using RDMS-FFD. The global deformation is shown in Fig.7(b). Fig.7(c) illustrates the local deformation of the results presented in Fig.7(b). The multiresolution deformation of the mouth in Fig.7(b) is shown in Fig.7(d). Fig.8 presents three deformation results of the bird model.

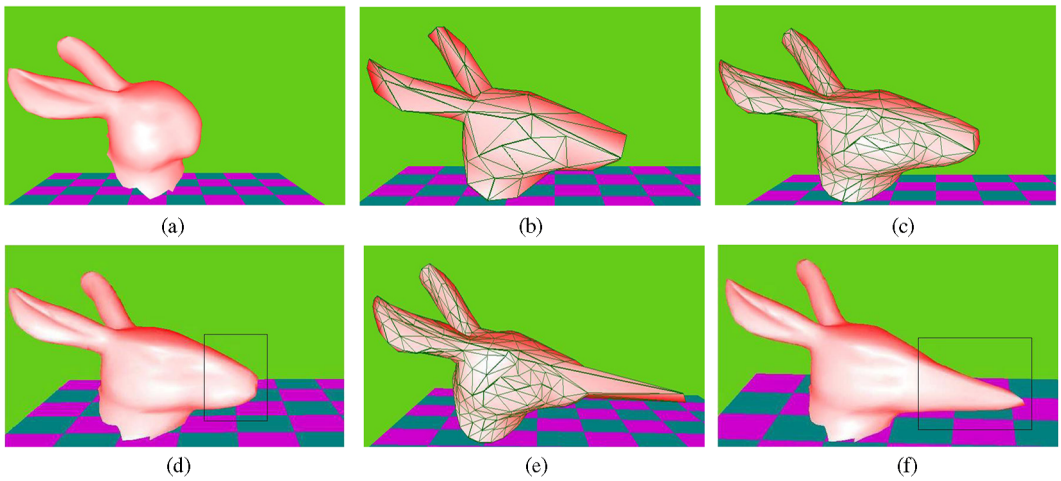

Fig.3. Multiresolution of the bunny head. (a) Initial bunny head model. (b) Deform tetrahedral domain. (c) Deform object by (b). (d) Refine domain in (b). (e) Deform the refined domain in (d). (f) Deform object by (e).

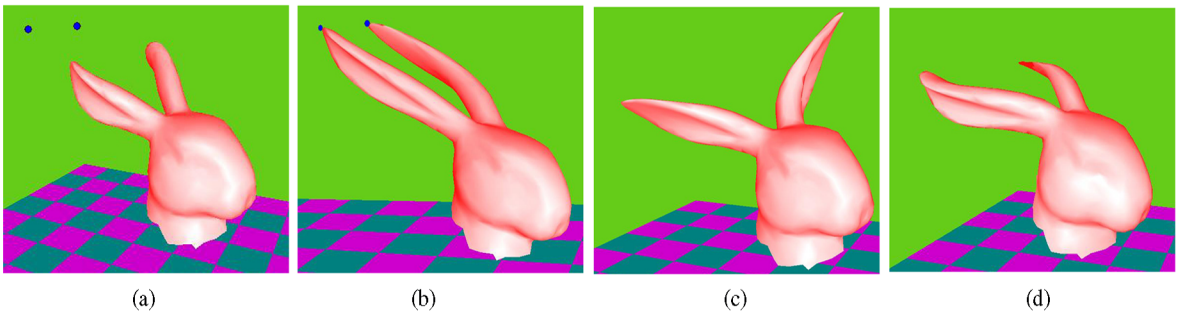

Fig.4. Direct manipulation of the bunny ear. (a) Initial bunny head model and two objective points. (b) Deformed object through objective points. (c)∼(d) Deformed object.

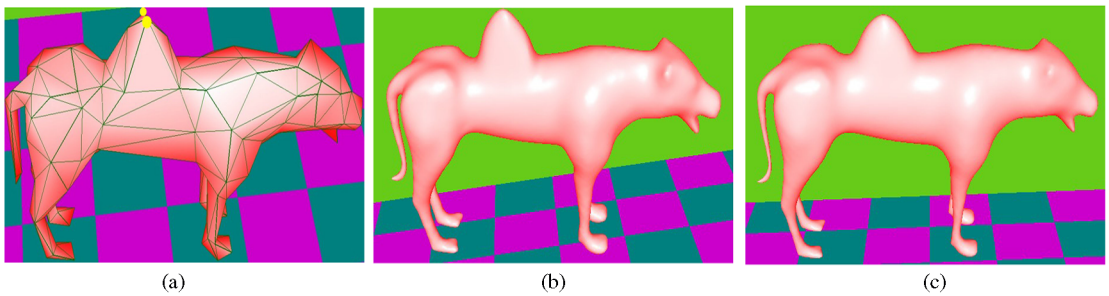

Fig.5. Effects of the weights. (a) Deformed tetrahedral domain. (b)∼(c) Deformation results with respect to different weights at the vertices with yellow color in (a).

We also evaluate the performance of RDMS-FFD technique by calculating the number of vertices of the tetrahedral domain and the parametrization time. The parametrization was measured on a PC with 3.0GHz Pentium-IV CPU, 1.0GB of RAM and Windows OS. The statistic numbers of the models together with the parametrization time are listed in Table 1. From Table 1, we can find that the number of the vertices of the tetrahedral domain in our method is smaller than the number of control points generated by octree subdivision for w-TFFD. So easy-to-use is improved in our method. The parametrization process is the most time-consuming step. As shown in Table 1, the parametrization time depends heavily on the number of the vertices of the original model. Fortunately, the parametrization time will not influence the subsequent deformation interaction.

Remark 1. Similar to the DMS-spline surfaces, the

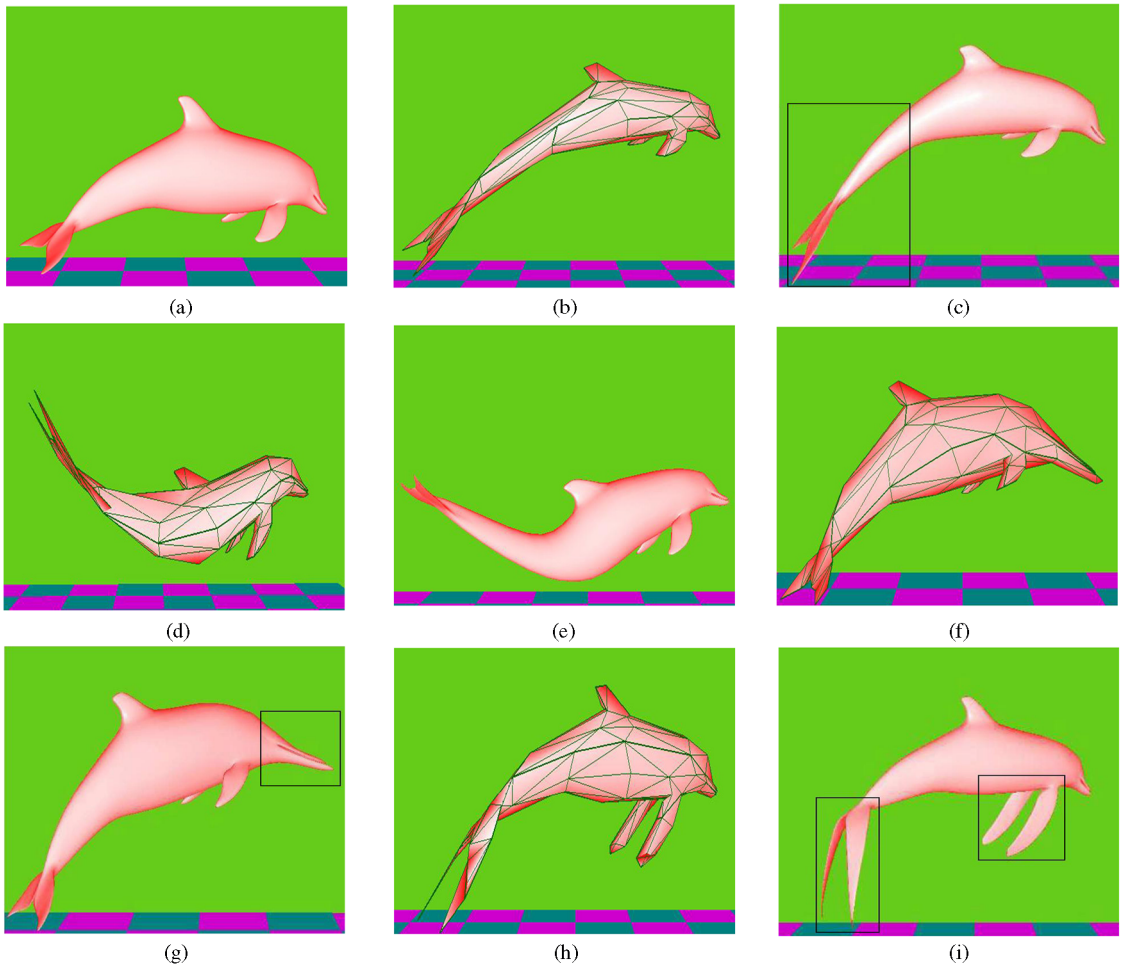

Fig.6. Deformation of the dolphin model. (a) Initial dolphin model. (b) Modified tetrahedral domain. (c) Deformed object by (b). (d) Modified tetrahedral domain. (e) Deformed object by (d). (f) Modified tetrahedral domain. (g) Deformed object by (f). (h) Modified tetrahedral domain. (i) Deformed object by (h).

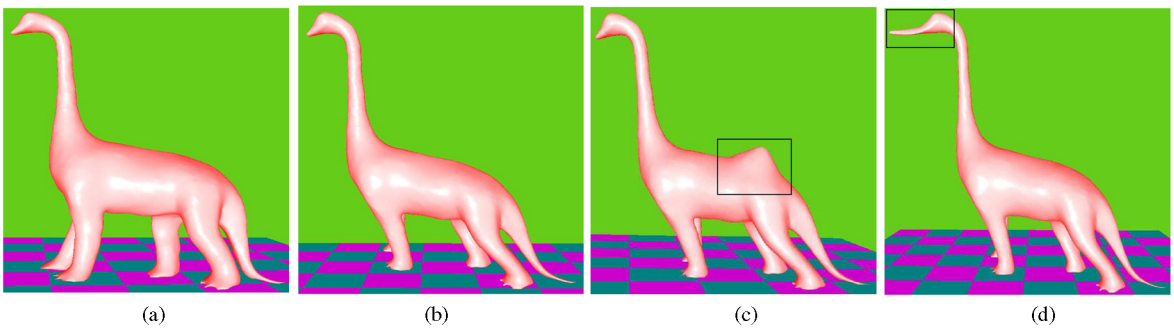

Fig.7. Deformation of the dinosaur model. (a) Initial dinosaur model. (b)∼(d) Deformed object.

RDMS-spline volume also has the so-called knot-lines. In order to eliminate these knot lines efficiently, we propose an approximate method based on the method in [39]. We minimize the integral of an approximation of the curvature along the lines in the definition domain:

$$\min \int_L (\boldsymbol{F}_{uu} + \boldsymbol{F}_{vv} + \boldsymbol{F}_{ww})^2 \mathrm{d}u\mathrm{d}v\mathrm{d}w, \qquad (10)$$

where L denotes the lines of the tetrahedral domain.

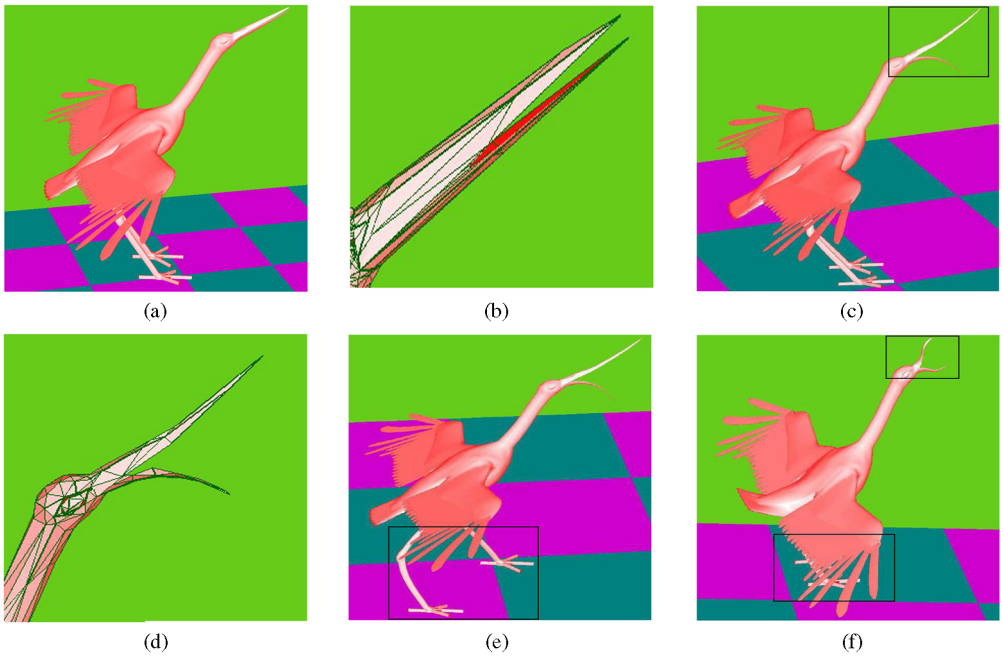

Fig.8. Deformation of the bird model. (a) Initial bird model. (b) Tetrahedral domain at the mouth. (c)∼(f) Deformed object.

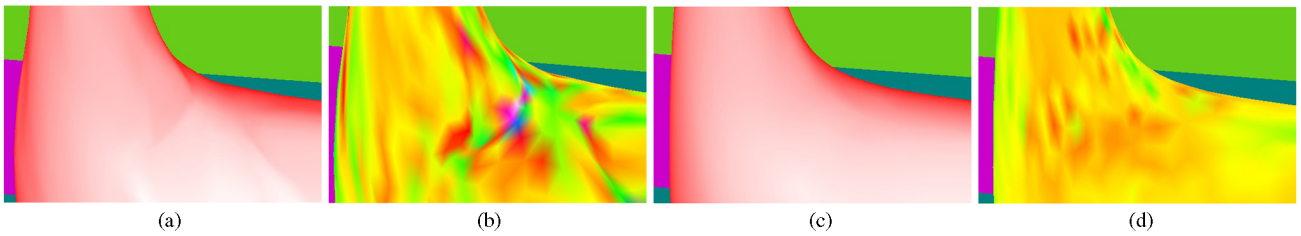

Fig.9. Knot-line elimination. (a)∼(b) Initial RDMS-spline volume and its mean curvature distribution. (c)∼(d) RDMS-spline volume after knot-line elimination and mean curvature distribution.

Note that the control points are fixed and only the weights are free variables in (10). We adopt the conjugate gradient method to solve this nonlinear optimization problem. The example of the dinosaur model is shown in Fig.9.

6 Conclusion and Future Work

In this paper, a novel free-form deformation technique is proposed. The technique is based on rational DMS-spline volume and inherits some good properties of rational DMS-spline volume. More importantly, it combines more advantages than previous FFD methods in a consistent frame work, such as local deformation, control lattice of arbitrary topology, smooth deformation, multiresolution deformation, and direct manipulation of deformation. Our experimental results for mesh models show that our method is powerful, flexible and intuitive for shape design in solid modeling and computer animation.

In the future, we will address some remaining limitations of the current method. Firstly, finding a tetrahedral domain of freeform objects is still an opening problem. Secondly, the main difficulty in our method is the parametrization process. Hence, finding an efficient parametrization method is an important problem to improve the current implementation. Finally, how to introduce hierarchy-based interactions and volume-preserving interactions as presented in [21] into RDMS-FFD is also very valuable.

References

[1] Dahmen W, Micchelli C A, Seidel H P. Blossoming begets B-spline built better by B-patches. *Mathematics of Computation*, 1992, 59(199): 97–115.

[2] Yan H, Hu S, Martin R. 3D morphing using strain field interpolation. *Journal of Computer Science and Technology*, 2007, 22(1): 147–155.
[3] Liu X, Bao H, Peng Q. Digital differential geometry processing. *Journal of Computer Science and Technology*, 2006, 21(5): 847–860.
[4] Sederberg T, Parry S. Free-form deformation of solid geometric models. In *Proc. ACM SIGGRAPH 1986*, Dallas, USA, 1986, pp.151–160.
[5] Bechmann D, Gain J. Point, curve, surface, volume-based spatial deformation. Technique Report, 2005, http://dpt-info.u-strasbg.fr/bechmann/Deformation1.pdf.
[6] Qin H, Terzopoulos D. Triangular NURBS and their dynamic generalizations. *Computer Aided Geometric Design*, 1997, 14(4): 325–347.
[7] Hua J, He Y, Qin H. Multiresolution heterogeneous solid modeling and visualization using trivariate simplex splines. In *Proc. ACM Symp. Solid Modeling and Application*, Nice, France, 2004, pp.47–58.
[8] Feng J, Shao J, Jin X, Peng Q, Forrest A R. Multiresolution free-form deformation with subdivision surface of arbitrary topology. *The Visual Computer*, 2006, 22(1): 28–42.
[9] Griessmair J, Purgathofer W. Deformation of solids with trivariate B-splines. In *Proc. Eurographics 1989*, Hamburg, Germany, 1989, pp.137–148.
[10] Kalra P, Mangli A, Thalmann N M, Thalmann D. Simulation of facial muscle actions based on rational freeform deformation. *Computer Graphic Forum*, 1992, 11(3): 59–69.
[11] Lamousin H, Waggenspack W. NURBS-based freeform deformation. *IEEE Computer Graphics and Application*, 1994, 14(9): 59–65.
[12] Raviv A, Elber G. Three dimensional freeform sculpting via zero sets of scalar trivariate functions. In *Proc. ACM Symp. Solid Modeling and Application*, Michigan, USA, 1999, pp.246–257.
[13] Kagan P, Fischer A. Integrated mechanically based CAE system using B-Spline finite elements. *Computer Aided Design*, 2000, 32(8): 539–552.
[14] Coquillart S. Extended free-form deformation: A sculpturing tool for 3D geometric modeling. In *Proc. ACM SIGGRAPH 1990*, Dallas, USA, 1990, pp.187–193.
[15] Coquillart S, Jancene P. Animated free-form deformations: An interactive animation technique. In *Proc. ACM SIGGRAPH 1991*, Las Vegas, USA, 1991, pp.23–26.
[16] MacCracken R, Joy K. Free-form deformation with lattice of arbitrary topology. In *Proc. ACM SIGGRAPH 1996*, New Orleans, USA, 1996, pp.181–188.
[17] Bechmann D, Bertrand Y, Thery S. Continuous free form deformation. *Comput. Netw. ISDN Syst.*, 1996, 29(14): 1715–1725.
[18] Moccozet L, Thalmann N M. Dirichlet free-form deformations and their application to hand simulation. In *Proc. Computer Animation*, Seoul, Korea, 1997, pp.93–102.
[19] Song W, Yang X. Free-form deformation with weighted T-spline. *The Visual Computer*, 2005, 21(3): 139–151.
[20] Ju T, Schaefer S, Warren J. Mean value coordinates for closed triangular meshes. In *Proc. ACM SIGGRAPH 2005*, Los Angeles, USA, 2005, pp.561–566.
[21] Yoon S H, Kim M S. Sweep-based freeform deformation. *Computer Graphics Forum*, 2006, 25(3): 487–496.
[22] Feng J, Ma L, Peng Q. A new free-form deformation through the control of parametric surfaces. *Computer and Graphics*, 1996, 20(4): 531–539.
[23] Feng J, Peng Q. Accurate B-spline free-form deformation of polygonal objects. *Journal of Graphics Tools*, 1998, 3(3): 11–27.
[24] Kazuya G, Kobayashi, Katsutoshi O. t-FFD: Freeform deformation by using triangular mesh. In *Proc. ACM Symp. Solid Modeling and Application*, Washington, USA, 2003, pp.226–234.
[25] Hua J, Qin H. Free form deformations via sketching and manipulating the scalar fields. In *Proc. ACM Symp. Solid Modeling and Application*, Washington, USA, 2003, pp.328–333.
[26] Lazarus F, Coquillart S, Jancene P. Axial deformations: An intuitive deformation technique. *Computer Aided Design*, 1994, 26(8): 608–612.
[27] Chang Y, Rockwood P. A generalized de Casteljau approach to 3D free-form deformation. In *Proc. ACM SIGGRAPH 1994*, Orlando, USA, 1994, pp.257–260.
[28] Singh K, Fiume E. Wires: A geometric deformation technique. In *Proc. ACM SIGGRAPH 1998*, Orlando, USA, 1998, pp.405–414.
[29] Hsu W, Hughes J, Kaufman H. Direct manipulation on free-form deformation. In *Proc. ACM SIGGRAPH 1992*, Chicago, USA, 1992, pp.177–184.
[30] Shimin Hu, Hui Zhang, Tai C, Jiaguang Sun. Direct manipulation of FFD: Efficient explicit solutions and decomposable multiple point constraints. *The Visual Computer*, 2001, 17(6): 370–379.
[31] Chang T, Lee J H, Kim M S, Hong S J. Direct manipulation of generalized cylinders based on rational motion. *The Visual Computer*, 1998, 15(5): 228–239.
[32] Aubert F, Bechmann D. Volume preserving space deformation. *Computer and Graphics*, 1997, 20(5): 625–639.
[33] Wang C, Kai Tang. Developability-preserved free-form deformation of assembled patches. In *Proc. ACM Symp. Solid Modeling and Application*, Nice, France, 2004, pp.231–236.
[34] Fong P, Seidel H P. An implementation of triangular B-spline surfaces over arbitrary triangulations. *Computer Aided Geometric Design*, 1993, 10(3/4): 267–275.
[35] Greiner G, Seidel H P. Modeling with triangular B-splines. *IEEE Computer Graphics and Applications*, 1994, 14(2): 56–60.
[36] Pfeifle R, Seidel H P. Faster evaluation of qudartic bivariate DMS spline surfaces. In *Proc. Graphics Interface 1994*, Alberta, Canada, 1994, pp.182–189.
[37] Franssen M, Veltkamp R C, Wesselink W. Efficient evaluation of triangular B-spline surfaces. *Computer Aided Geometric Design*, 2000, 17(9): 863–877.
[38] He Y, Qin H. Surface reconstruction with triangular B-splines. In *Proc. Geometric Modeling and Processing 2004*, Beijing, China, 2004, pp.279–291.
[39] He Y, Gu X, Qin H. Rational spherical splines for genus zero shape modeling. In *Proc. Shape Modeling and Applications 2005*, Cambridge, MA, USA, 2005, pp.82–91.
[40] He Y, Gu X, Qin H. Automatic shape control of triangular B-splines of arbitrary topology. *Journal of Computer Science and Technology*, 2006, 21(2): 232–237.
[41] Gu X, He Y, Qin H. Manifold splines. In *Proc. ACM Symp. Solid and Physical Modeling* (*SPM 05*), Cambridge, Massachusetts, USA, 2005, pp.27–38.
[42] Sander P V, Gu X, Gortler S J, Hoppe H, Snyder J. Silhouette clipping. In *Proc. ACM SIGGRAPH 2000*, New Orleans, USA, 2000, pp.327–334.
[43] Zhou K, Huang J, Snyder J, Liu X, Bao H, Guo B, Shum H Y. Large mesh deformation using the volumetric graph Laplacian. *ACM Transactions on Graphics*, 2005, 24(3): 496–503.
[44] Huang J, Shi X, Liu X, Zhou K, Wei L, Teng S, Bao H, Guo B, Shum H Y. Subspace gradient domain mesh deformation. *ACM Transactions on Graphics*, 2006, 25(3): 1126–1134.
[45] Jorge N, Stephen J W. Numerical Optimization. Berlin, Heidelberg, New York: Springer, 1999.

[46] Schaefer S, Hakenberg J, Warren J. Smooth subdivision of tetrahedral meshes. In *Proc. Eurographics Symp. Geometry Processing (SGP 04)*, Nice, France, 2004, pp.151–158.

Gang Xu will join INRIA Sophia-Antipolis as a postdoctoral fellow in October, 2008. He received his Bachelor's degree in computational mathematics from Shandong University in 2003 and Ph.D. degree in applied mathematics from Zhejiang University in 2008. His research interests include computer aided geometric design, digital geometry processing, and image processing.

Guo-Zhao Wang is a professor of applied mathematics in Zhejiang University. His main research interests include computer aided geometric design, computer graphics and medical image processing.

Xiao-Diao Chen is a faculty member in College of Computer, Hangzhou Dianzi University. He received his B.S. degree in mathematics from Zhejiang University in 1996 and Ph.D. degree in computer science from Tsinghua University in 2006, respectively. His main research interests include computer aided design and computer graphics.

Correction

The 4th author's name of the paper entitled "Middleware for Wireless Sensor Networks: A Survey", published on No.3, 2008, pp.305–326, should be Sajal K. Das.