Visual Comput (2007) 23: 1027–1033
DOI 10.1007/s00371-007-0170-3

ORIGINAL ARTICLE

Xin Li
Jiansong Deng
Falai Chen

Surface modeling with polynomial splines over hierarchical T-meshes

Published online: 8 August 2007

Xin Li · Jiansong Deng (✉) · Falai Chen
Department of Mathematics
University of Science and Technology of China
Hefei, Anhui 230026, P.R. China
{dengjs, chenfl}@ustc.edu.cn

Abstract Computer graphics and computer-aided design communities prefer piecewise spline patches to represent surfaces. But keeping the smoothness between the adjacent patches is a challenging task. In this paper, we present a method for stitching several surface patches, which is a key step in complicated surface modeling, with polynomial splines over hierarchical T-meshes (PHT-spline for short). The method is simple and can be easily applied to complex surface modeling. With the method, spline surfaces can be constructed efficiently and adaptively to fit genus-zero meshes after their spherical parameterization is obtained, where only small sized linear systems of equations are involved.

Keywords Surface modeling · Spline · Hierarchical T-mesh · Stitching

1 Introduction

In geometric modeling, the representations of surfaces is one of the most important and interesting research topics. Computer graphics and computer-aided design communities prefer parametric surfaces, especially spline surfaces, since their representations are simple, and points on the surface can be easily determined [5]. These kinds of surfaces are commonly supported within current surface modeling systems. The commonly used spline surfaces are tensor product (TP) B-splines or triangular splines. With triangular B-splines, one must face the complexity of dimension calculation and basis function construction in the spline space. In fact, there are still many theoretical issues under research for triangular spline spaces [11]. Surface fitting with TP B-spline surfaces needs to overcome the weakness that the control points must lie topologically in a rectangular grid. To model a geometric object with complex topology, one has to smoothly piece many B-spline patches together. But keeping the smoothness between the adjacent patches is a challenging task. Most methods [3, 4] restrict all the patches to have the same knot vectors and same degrees. In the current CAD/CAM software systems, such as Maya and 3D Max, their stitching tools also have this limit. Figure 1 shows an ear model which is composed of 457 B-spline patches, where the adjacent patches have many gaps when the knot vectors of the adjacent patches do not match. The bottom part of Fig. 1 shows the gap among three of the patches. In this paper, we present a method for stitching several surface patches with a PHT-spline – a polynomial spline over a hierarchical T-mesh. The method is simple and can be easily used in complex models. The bottom pictures of Fig. 1 depict the result from our stitching algorithm with C^0 continuity.

Geometric models are often described as closed, genus-zero meshes. In this paper, we construct a PHT-spline to approximate genus-zero mesh models. In the process, we first partition the meshes into several parts, fitting each part with a PHT-spline surface. Then we stitch them into one surface patch. The process is very efficient

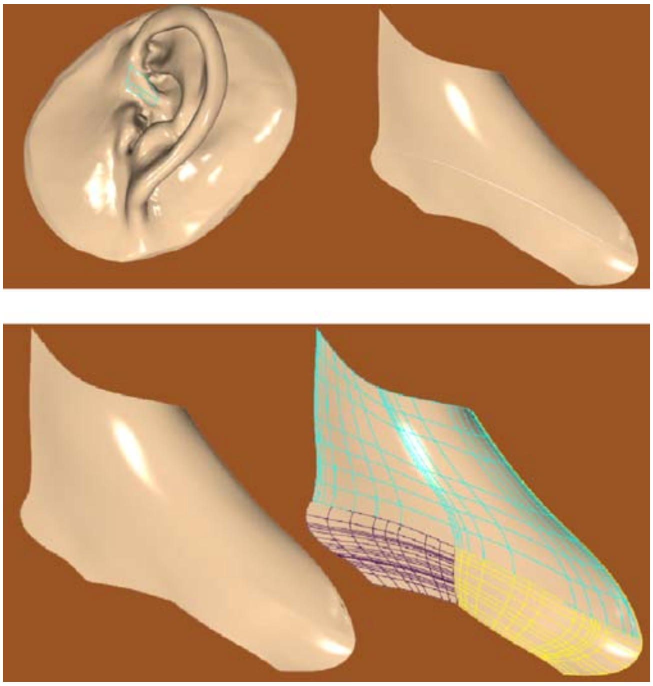

Fig. 1. A gap in the ear model fixed with C^0 continuity

and adaptive and it is very easily to extend to an arbitrary topology.

The remainder of the paper is organized as follows: Section 2 reviews some related work. Section 3 recalls some preliminary knowledge about PHT-splines. Section 4 proposes an algorithm for stitching surface patches with PHT-splines. Section 5 provides a PHT-spline surface fitting algorithm for genus-zero meshes. Section 6 concludes the paper with a summary and future work.

2 Related work

The literature on local control refinement of B-spline surfaces was initiated by Forsey and Bartels. They invented hierarchical B-splines and introduced two concepts: local refinement using an efficient representation and multi-resolution editing [6]. In 2003 and 2004, Sederberg et al. [9, 10] introduced the notion of T-splines, where many valuable operations are provided as well. But some of them are not simple, such as control point insertion whose complexity is uncertain in some cases. They also provided the algorithm for merging B-splines into T-splines, which is based on the control point insertion algorithm. In [1, 2], polynomial spline functions over T-meshes are introduced, where the spline function on every cell is a TP polynomial, and achieves the specified smoothness across the common edges. The new splines not only inherit the main advantage of Sederberg's T-splines, adaptivity, but also have the advantages over Sederberg's T-splines in many aspects. For example, the new splines are a single polynomial in each mesh cell, which will reduce the time-cost in many geometric operations.

3 Polynomial splines over hierarchical T-meshes

A T-mesh is basically a rectangular grid that allows T-junctions [1, 9]. Given a T-mesh $\mathcal{T}$, let $\mathcal{F}$ denote all the cells in $\mathcal{T}$ and Ω the region occupied by all the cells in $\mathcal{T}$,

$$\mathcal{S}(m, n, \alpha, \beta, \mathcal{T}) := \left\{ s(x, y) \in C^{\alpha,\beta}(\Omega) \middle| s(x, y)|_\phi \in \mathrm{P}_{mn} \text{ for any } \phi \in \mathcal{F} \right\},$$

where P_{mn} is the space of all the polynomials with bi-degree (m, n), and $C^{\alpha,\beta}(\Omega)$ is the space consisting of all the bivariate functions which are continuous in Ω, with order α along the x-direction and with order β along the y-direction. It is obvious that $\mathcal{S}(m, n, \alpha, \beta, \mathcal{T})$ is a linear space, which is called the spline space over the given T-mesh $\mathcal{T}$. As discussed in [1], the dimension of the spline space $\mathcal{S}(2\alpha+1, 2\beta+1, \alpha, \beta, \mathcal{T})$ is $(\alpha+1)(\beta+1)$ times the number of crossing vertices and boundary vertices (which are later called basis vertices).

A hierarchical T-mesh is a special type of T-mesh, which has a natural level structure. Level-0 mesh is the standard TP mesh. If level-i mesh is given, then the level-$i+1$ mesh is obtained by subdividing some of the cells in level-i. Each cell is subdivided into four subcells by connecting the middle points of the opposite edges in the cell.

In [2], the present authors proposed a method to construct basis functions of $\mathcal{S}(3, 3, 1, 1, \mathcal{T})$ over hierarchical T-meshes. The basis functions are constructed level by level, and have many good properties, such as non-negativity, compact support and partition of unity.

For any function $b(u, v)$, its function value $b(u, v)$, two partial derivatives of first order and mixed partial derivative are

$$b_u(u, v) = \frac{\partial}{\partial u} b(u, v)|_{(u,v)},$$
$$b_v(u, v) = \frac{\partial}{\partial v} b(u, v)|_{(u,v)},$$
$$b_{uv}(u, v) = \frac{\partial^2}{\partial u \partial v} b(u, v)|_{(u,v)}.$$

At some point (u_0, v_0) are called the geometric information of $b(u, v)$ at point (u_0, v_0).

Given a hierarchical T-mesh $\mathcal{T}$, suppose the basis functions are $\{b_j^k(u, v)\}$, $j = 1, \ldots, N$, $k = 0, \ldots, 3$. Here N is the number of basis vertices. Then a spline surface over $\mathcal{T}$

can be defined as

$$\boldsymbol{S}(u,v)=\sum_{j=1}^{N}\sum_{k=0}^{3}\boldsymbol{C}_j^k b_j^k(u,v), \quad (1)$$

where $\boldsymbol{C}_j^k$ are the control points associated with the jth basis vertex.

Suppose the associated domain of the ith basis vertex is $(u_i^2, u_i^1, v_i^2, v_i^1)$. We can then write

$$(\boldsymbol{B}_i)=\begin{pmatrix} 1 & 1 & 1 & 1 \\ -\frac{u_i}{\alpha} & \frac{1-u_i}{\alpha} & -\frac{u_i}{\alpha} & \frac{1-u_i}{\alpha} \\ -\frac{v_i}{\beta} & -\frac{v_i}{\beta} & \frac{1-v_i}{\beta} & \frac{1-v_i}{\beta} \\ \frac{u_i v_i}{\alpha\beta} & -\frac{(1-u_i)v_i}{\alpha\beta} & -\frac{u_i(1-v_i)}{\alpha\beta} & \frac{(1-u_i)(1-v_i)}{\alpha\beta} \end{pmatrix},$$

where $u_i=\frac{u_i^2}{u_i^1+u_i^2}$ and $v_i=\frac{v_i^2}{v_i^1+v_i^2}$. Assume the geometric information at the ith basis vertex are f, f_u, f_v, f_{uv}, respectively. Then the control points of the ith basis vertex are computed as

$$\left(\boldsymbol{C}_i^0, \boldsymbol{C}_i^1, \boldsymbol{C}_i^2, \boldsymbol{C}_i^3\right)=(f, f_u, f_v, f_{uv})(\boldsymbol{B}_i). \quad (2)$$

In [2], an efficient scheme is proposed to fit scattered data $\{\boldsymbol{P}_i\}_{i=1}^N$ with spline surfaces over hierarchical T-meshes. According to Eq. 2, the main approach is to estimate the geometric information at all the basis vertices. The basic idea is as follows: First, estimate the geometric information at every point $\boldsymbol{P}_i$. For every point $\boldsymbol{P}_i$ in the given mesh, its topological neighborhood $\mathcal{N}(\boldsymbol{P}_i)$ is organized with enough points for information estimation. Then fit a bi-cubic or bi-quadratic patch to the points in $\mathcal{N}(\boldsymbol{P}_i)$ (assuming a parameterization of the mesh model is obtained). The required geometric information at point $\boldsymbol{P}_i$ is obtained by evaluating the patch at the corresponding parameter value of $\boldsymbol{P}_i$. Second, for each basis vertex $\boldsymbol{Q}$, the geometric information at $\boldsymbol{Q}$ can be obtained by linearly interpolating the geometric information of three neighboring points $\boldsymbol{P}_i$, $\boldsymbol{P}_j$ and $\boldsymbol{P}_k$ of $\boldsymbol{Q}$. For a detailed description of the basis construction and fitting process, the reader is referred to [2].

4 Stitching surface patches

This section discusses the algorithm for stitching several PHT-spline surface patches. In the algorithm, we only update the information near the common boundary curves. Because a polynomial B-spline surface is a special kind of PHT-spline, the algorithm is also applicable for stitching B-spline surface patches.

4.1 Continuity conditions

Let $\pi_1(x,y)$ and $\pi_2(x,y)$ be two bi-cubic polynomials defined over two adjacent domains $[x_0,x_1]\times[y_0,y_1]$ and $[x_1,x_2]\times[y_0,y_1]$, respectively. They can be expressed in the Bernstein–Bézier forms, with Bézier ordinates $\{b_{j,k}^1\}$ and $\{b_{j,k}^2\}$, respectively.

It is well known that $\pi_1(x,y)$ and $\pi_2(x,y)$ are C^1 continuous across their common boundary if and only if

$$b_{3,j}^1=b_{0,j}^2 \quad j=0,\dots,3,$$

$$\frac{b_{3,j}^1-b_{2,j}^1}{x_1-x_0}=\frac{b_{1,j}^2-b_{0,j}^2}{x_2-x_1} \quad j=0,\dots,3,$$

where the first equation is the condition for C^0 continuity.

The geometric continuity conditions for $\pi_1(x,y)$ and $\pi_2(x,y)$ are as follows: For G^0 continuity,

$$\pi_1(x_1,y)=\pi_2(x_1,y), \quad y\in[y_0,y_1]. \quad (3)$$

For $\pi_1(x,y)$ and $\pi_2(x,y)$ to be G^1 continuous, we require, in addition,

$$\frac{\partial}{\partial x}\pi_1=p(y)\frac{\partial}{\partial x}\pi_2+q(y)\frac{\partial}{\partial y}\pi_1, \quad (4)$$

for $x=x_1$, $y\in[y_0,y_1]$ and some functions $p(y)$ and $q(y)$, with $p(x)>0$; here $p(x)$ and $q(x)$ are called the connecting functions.

4.2 Continuity conditions for PHT surfaces

The continuity conditions for two PHT-spline surfaces are illustrated with an example in Fig. 2. Suppose two PHT-spline surfaces $\boldsymbol{S}^1(u,v)$ and $\boldsymbol{S}^2(u,v)$ over T-meshes $\mathcal{T}^1$ and $\mathcal{T}^2$ are given, where $\mathcal{T}^1$ and $\mathcal{T}^2$ share a common boundary line (but the two boundary line segments may not exact coincide). We consider a basis vertex $A\in\mathcal{T}^1$ on the common part of the boundary lines. Suppose $B\in\mathcal{T}^2$ is the nearest vertex to A. If B is not a basis vertex, then its geometry information is determined by its neighbor basis vertices in $\mathcal{T}^2$. This leads to discontinuity along the common boundary. Hence a knot line segment should be inserted through B in a fashion as shown in Fig. 2. Similarly, other knot segments are inserted in $\mathcal{T}^1$ or $\mathcal{T}^2$ when running through the boundary basis vertices in both meshes. Consequently, on the common part of the boundary lines, every boundary basis vertex in one mesh has a corresponding boundary basis vertex in the other mesh, as shown in Fig. 2.

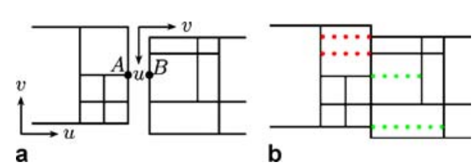

Fig. 2a,b. Continuity conditions for spline surfaces over T-meshes

Now we can use the geometric information to express the continuity conditions. The C^0 continuity conditions across their common boundary are at each of the basis vertices A in $\mathcal{T}^1$ and the corresponding basis vertex B in $\mathcal{T}^2$, then the following equations hold:

$$\boldsymbol{S}^1(A) = \boldsymbol{S}^2(B); \quad \boldsymbol{S}^1_v(A) = -\boldsymbol{S}^2_u(B).$$

While the C^1 continuity conditions across their common boundary are, in addition, as follows:

$$\boldsymbol{S}^1_u(A) = \boldsymbol{S}^2_v(B); \quad \boldsymbol{S}^1_{uv}(A) = -\boldsymbol{S}^2_{uv}(B).$$

4.3 Stitching surface patches

In geometric modeling, an object usually consists of many parts, each of which is possibly modeled independently with different B-spline surface patches. Two neighbor surfaces might have different knot vectors along their common boundary, such as the ear model in Fig. 1. As shown in Fig. 3a,b, the grey and the black control grids are defined over the different knot vectors. Stitching them into a single B-spline surface requires that the two surfaces have the same common knot vector along the common boundary lines. Hence knot insertion must first be performed before stitching. However, in a TP spline surface, these knot insertions can significantly increase the number of control points.

But stitching with PHT-splines is different. Here we only need to modify a narrow band of the surfaces along their common boundary curves. The main steps of the stitching algorithm are as follows:

Step 1: Determine the common parameter domains. The first step is to find the common parameter boundary domains of the two patches. As illustrated in Fig. 4, at first we extract B-spline curves which form the boundaries of the two surfaces. Then, find a curve segment on each boundary curve such that the two curve segments (say, AB and CD in Fig. 4) are near to each other within a given tolerance. Finally, we map the two curve segments back to get the parameter domains.

For simplicity, the common parameter domains can also be specified by users.

Step 2: Reparameterize surfaces. This step is to reparameterize the two surfaces such that the parameterization of

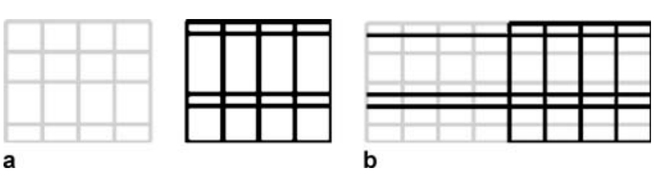

Fig. 3a,b. Stitching with B-splines

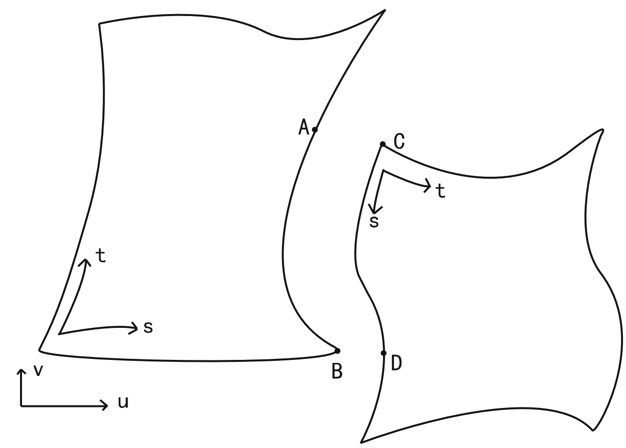

Fig. 4. Common boundary

the two surfaces coincides along the common boundary curve.

As illustrated in Fig. 4, suppose the surface on the right of Fig. 4 has geometric information is f, f_s, f_t, f_{st}, then we should convert them to g, g_u, g_v, g_{uv} with

$$g = f; \quad g_u = f_t; \quad g_v = -f_s; \quad g_{uv} = -f_{st}.$$

This is very important for Steps 3 and 4.

Next is to reparameterize the two surfaces to make them have the same knot values along the common boundary. Here we use the linear reparameterization with function $f(t) = at + b$ along the common boundary. In order to guarantee that the common domains have coincident parameter (i.e., the parameters at A and C are the same, also at B and D), we can use these conditions to determine the constants a and b.

Step 3: Combine the boundary information. In order for that, along the common boundary a basis vertex in one patch is also a basis vertex in the other patch, we should insert some knot segments into T-meshes (see Fig. 3 as an example). The geometric information at these basis vertices is the average of the geometric information at the two basis vertices on the common boundary (red points in Fig. 3).

For example, in Fig. 2, suppose the geometric information at A and B are $(f^1, f^1_s, f^1_t, f^1_{st})$ and $(f^2, f^2_s, f^2_t, f^2_{st})$, respectively.

For C^0 stitching, the new geometric information at A and B should be $(f, f^1_s, f_v, f^1_{st})$ and $(f, -f_v, f^2_t, f^2_{st})$, respectively. Here,

$$f = \frac{f^1 + f^2}{2}; \quad f_v = \frac{f^1_t - f^2_s}{2}.$$

For C^1 stitching, the new geometric information at A and B should be (f, f_u, f_v, f_{uv}) and $(f, -f_v, f_u, -f_{uv})$, respectively. Here,

$$f = \frac{f^1 + f^2}{2}; \quad f_u = \frac{f_s^1 + f_t^2}{2};$$
$$f_v = \frac{f_t^1 - f_s^2}{2}; \quad f_{uv} = \frac{f_{st}^1 - f_{st}^2}{2}.$$

Step 4: Interpolate the information. Applying Eq. 2, we can easily obtain the control points for the PHT-splines.

There are mainly three cases in the stitching procedure which are illustrated in Fig. 5. The first case is to stitch two surfaces which have an aligned common boundary, as shown in Fig. 5a. In this case, we just need to apply the former algorithm directly. The pictures in Fig. 6a are the stitching results for this case. The colored curves are the isoparameter curves projected onto the surfaces.

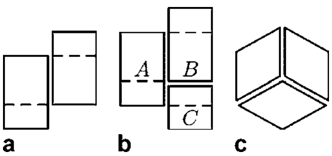

Fig. 5a–c. The three cases for stitching

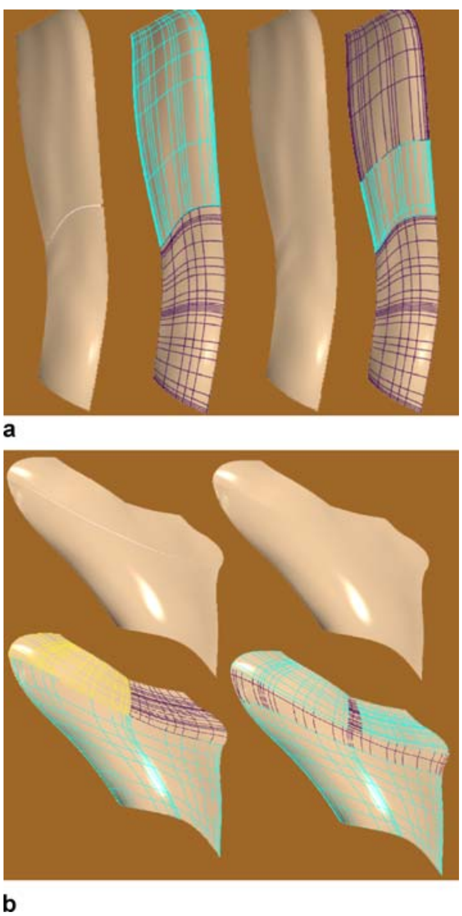

Fig. 6a,b. Stitching two pieces and three pieces of surface patches

The second case is to stitch three patches, illustrated in Fig. 5b. In this case, one patch shares a boundary with the other two patches. We can apply the former algorithm to stitch patches A and B, patches B and C, and patches C and A in some order. The pictures in Fig. 6b are the results for this case.

The third case is to stitch n patches around an extraordinary vertex, which has valence not equal to four. In this case, we cannot achieve global C^1 continuity when $n \neq 4$. The next section will provide the method of determining geometric information in detail. However, if $n = 4$, we can stitch the surfaces in the same fashion as the second case.

4.4 Stitching at extraordinary vertex

First we will review the geometric continuity conditions of order one provided in [8]. In fact, in Eq. 4, let $p(x) = 1$ and $q(x) = \alpha x + \beta(1 - x)$, then the following equations are sufficient conditions for two bi-cubic surfaces as shown in Fig. 7a to be G^1:

$$\begin{aligned}
&p_0 - q_0 = q_0 - r_0 + \alpha(q_0 - q_1);\\
&p_1 - q_1 = q_1 - r_1 + \frac{2}{3}\alpha(q_1 - q_2) + \frac{1}{3}\beta(q_0 - q_1);\\
&p_2 - q_2 = q_2 - r_2 + \frac{1}{3}\alpha(q_2 - q_3) + \frac{2}{3}\beta(q_1 - q_2);\\
&p_3 - q_3 = q_3 - r_3 + \beta(q_2 - q_3).
\end{aligned} \tag{5}$$

Here we call α the geometric constant for vertex q_0 along the common boundary and β the constant for vertex q_3.

If we set $p(x) = 1$ and $q(x) = \alpha(1 - x)^2$, which would automatically force the boundary curves to be quadratic, as shown in the last equation of Eq. 6, then the conditions are:

$$\begin{aligned}
&p_0 - q_0 = q_0 - r_0 + \alpha(q_0 - q_1);\\
&p_1 - q_1 = q_1 - r_1 + \frac{2}{3}\alpha(q_1 - q_2) - \frac{1}{3}\alpha(q_0 - q_1);\\
&p_2 - q_2 = q_2 - r_2;\\
&p_3 - q_3 = q_3 - r_3;\\
&q_0 - q_3 = 3(q_1 - q_2).
\end{aligned} \tag{6}$$

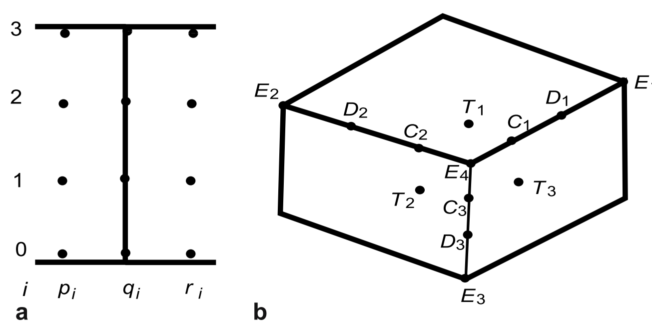

Fig. 7a,b. The G^1 continuity between two bi-cubic patches and around a common vertex

We will apply these two conditions into the stitching algorithm around an extraordinary vertex.

First we will discuss how to stitch at an extraordinary vertex of valence three. Suppose the Bézier control points are illustrated in Fig. 7b. In the figure, E_4 is an extraordinary vertex, but E_i, $i = 1, 2, 3$ are not extraordinary vertices. We can otherwise refine the three patches such that they are not extraordinary by inserting some knot lines. We can denote the geometric constant for vertex E_4 along E_iE_4 as α_i.

Notice that E_i, $i = 1, 2, 3$ are not extraordinary vertices, so the control points D_i and E_i should be kept unchanged. Thus here we apply Eq. 6 to describe the continuity conditions. That is to say we have to force the boundary curves to be quadratic, i.e.,

$$E_4 - E_i = 3(C_i - D_i).$$

Let $P_i = (3D_i - E_i)/2$, then $C_i = (2P_i - E_4)/3$. According to the first line of Eq. 6, we have:

$$C_{k+1} - E_4 = E_4 - C_{k-1} + \alpha_k(E_4 - C_k)$$

where $k = 1, 2, 3$, and $C_{k+3} = C_k$. Adding all three equations with $\alpha_i = \lambda$, we then determine that $E_4 = \frac{P_1+P_2+P_3}{6}$ and λ must be 1.

Now only the control points T_i should be computed. In fact, T_i is the unique solution of the second line of Eq. 6:

$$T_1 = \frac{3}{2}(D_1 + D_2 - D_3) - \frac{1}{3}(D_1 + D_2 - D_3) - \frac{E_4}{6},$$
$$T_2 = \frac{3}{2}(D_2 + D_3 - D_1) - \frac{1}{3}(D_2 + D_3 - D_1) - \frac{E_4}{6},$$
$$T_3 = \frac{3}{2}(D_3 + D_1 - D_2) - \frac{1}{3}(D_3 + D_1 - D_2) - \frac{E_4}{6}.$$

After having computed all the Bézier control points, we can compute the geometric information at the extraordinary vertex, using Eq. 2 to compute the control points for the PHT-spline.

If the valence of the extraordinary vertex is n, $n \neq 3, 4$, we can apply the similar method to compute the Bézier control points around the vertex (see [8] for details).

5 Fitting genus-zero meshes

Geometric models are often described by closed meshes with genus-zero. For such models, the sphere is the most natural parameterized domain. In this section, an assembly of spline surfaces over hierarchical T-meshes are constructed based on the spherical parameterization [7] with the stitching algorithm. The method is efficient and adaptive, where only small sized linear systems of equations are involved.

First, we partition the meshes into six parts according to the spherical parameterization. To do so, one selects an inscribed cube of the parameterized sphere, and maps the twelve edges of the cube onto the sphere with the central projection from the center of the sphere. These twelve curves on the sphere partition the mesh into six parts. Each parameterized part is central-projected onto the corresponding face of the cube to obtain its planar parameterization. Then we construct six PHT-spline surfaces to fit the six parts using the fitting algorithm in [2] level by level. Finally, we use the stitching method to stitch the six PHT-spline patches together to achieve C^1 continuity, except the patches around the extraordinary vertices, where G^1 is achieved.

The surface fitting algorithm mainly has the following steps:

1. Partition the given closed mesh into six parts according to its spherical parameterization and central projection. Then we obtain a planar parameterization for each face.
2. Construct a PHT-spline surface to fit each part of the mesh model using the fitting algorithm in [2].
3. Use the stitching method to stitch the six PHT-spline patches together.

It should be noted that this approach is different from what we did in [2], where, in order to maintain smoothness between any two neighbor patches, one needs, in each step of the fitting algorithm, to adjust the geometric informa-

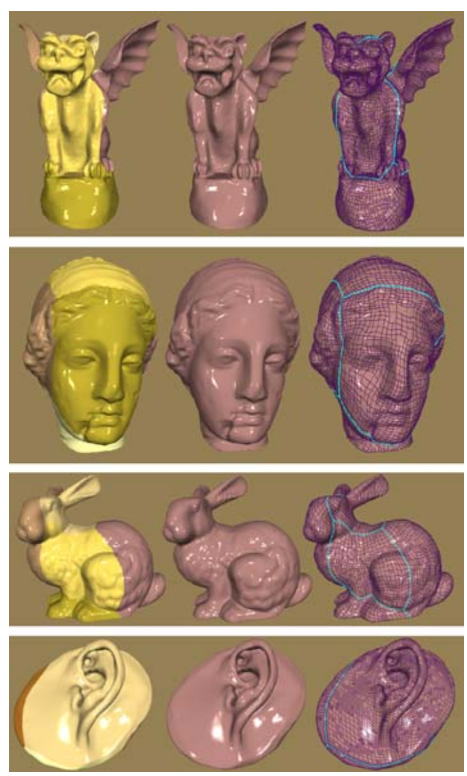

Fig. 8. Fitting examples

tion calculation along their boundary curves. This makes the fitting algorithm more complex.

Several examples are provided in Fig. 8 to illustrate the stitching algorithm. In each of the examples, the first image is the original mesh model which is partitioned into six parts. The second image is the fitting result, and the last is the result with parameter uv curves. The blue curves are the boundary curves of the six parts of the mesh model.

6 Conclusions and future work

The paper presents a method for stitching several PHT surface patches. The stitching method plays an important role in complex surface modeling. As an application, we apply this method in constructing PHT-spline surfaces to fit genus-zero meshes.

There are a number of issues for future research. For example, we only discuss surface fitting of genus-zero meshes in this paper. In the future, we will discuss how to construct several polynomial spline surfaces over hierarchical T-meshes to fit arbitrary topology triangle meshes.

Acknowledgement The authors were supported by A National Key Basic Research Project of China (No. 2004CB318000), the Outstanding Youth Grant of NSF of China (No. 60225002), NSF of China (No. 60533060, 60473132 and 10626049), Doctorial Program of MOE of China, and the 111 Project (No. b07033).

References

1. Deng, J., Chen, F., Feng, Y.: Dimensions of spline spaces over T-meshes. J. Comput. Appl. Math. **194**(1–2), 267–283 (2006)
2. Deng, J., Chen, F., Li, X., et al.: Polynomial splines over hierarchical T-meshes. Graph. Models (2006, in press) http://staff.ustc.edu.cn/~dengjs/files/papers/pht.pdf
3. Eck, M., Hoppe, H.: Automatic reconstruction of B-spline surfaces of arbitrary topological type. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, pp. 325–334. ACM, Boston (1996)
4. Eck, M., DeRose, T., Duchamp, T., Hoppe, H., Lounsbery, M., Stuetzle, W.: Multiresolution analysis of arbitrary meshes. In: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 173–182. ACM, Boston (1995)
5. Farin, G.: Curves and Surfaces for CAGD: A Practical Guide, 5th edn. Kaufmann, San Francisco (2002)
6. Forsey, D., Bartels, R.H.: Hierarchical B-spline refinement. Comput. Graph. **22**(4), 205–212 (1988)
7. Gotsman, C., Gu, X., Sheffer, A.: Fundamentals of spherical parametrization for 3d meshes. ACM Trans. Graph. **22**(3), 358–363 (2003)
8. Lee, S.L., Majid, A.A.: Closed smooth piecewise bicubic surfaces. ACM Trans. Graph. **10**(4), 342–365 (1991)
9. Sederberg, T.W., Zheng, J., Bakenov, A., Nasri, A.: T-splines and T-NURCCs. ACM Trans. Graph. **22**(3), 161–172 (2003)
10. Sederberg, T.W., Cardon, D.L., Finnigan, G.T., North, N.S., Zheng, J., Lyche, T.: T-spline simplification and local refinement. ACM Trans. Graph. **23**(3), 276–283 (2004)
11. Wang, R.H.: Multivariate Spline Functions and their Applications. Kluwer, Dordrecht (2001)

XIN LI was born in Hubei, China, in 1980. He was a Ph.D. candidate of University of Science and Technology of China. Currently he is a visiting scholar of Department of Compute Science in Brigham Young University. His research interests include computer-aided geometric design and computer graphics.

J. DENG was born in Shangdong, China, in 1971. He received his Ph.D. degree from the University of Science and Technology of China in 1998. Currently he is a professor in the Department of Mathematics at the University of Science and Technology of China. His research interests include computer-aided geometric design and computer graphics.

F. CHEN was born in Anhui, China, in 1966. He received his Ph.D. degree from the University of Science and Technology of China in 1994. He is a professor in the Department of Mathematics at the University of Science and Technology of China. His research interests include computer-aided geometric design and computer graphics.