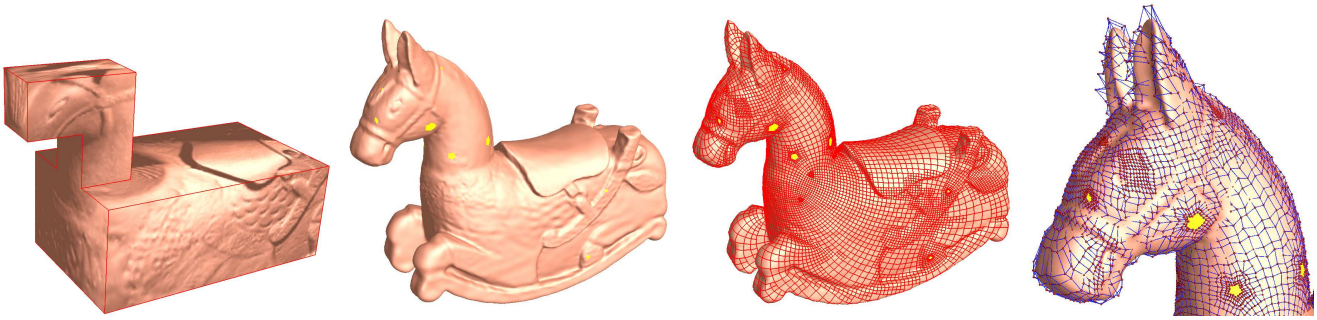
Polycube Splines

Hongyu Wang* Stony Brook
Ying He† NTU
Xin Li‡ Stony Brook
Xianfeng Gu§ Stony Brook
Hong Qin¶ Stony Brook



(a) Conformal polycube map (b) Polycube T-spline (c) T-junctions on polycube spline (d) Close-up of control points

Figure 1: Polycube spline for the Isidore Horse model. (a) The conformal polycube map serving as the parametric domain. (b) and (c) Polycube T-splines obtained via affine structure induced by the polycube map. Note that our polycube spline is globally defined as a "one-piece" shape representation without any cutting and gluing work except at the finite number of extraordinary points (corners of the polycube). The extraordinary points are colored in yellow in (b) and (c). The red curves on the spline surface (see (c)) highlight the T-junctions. (d) Close-up of the spline model overlaid with the control points. The polycube T-spline contains 12158 control points. The original model contains $150K$ vertices. The root-mean-square error is 0.07% of the diagonal of the model.

Abstract

This paper proposes a new concept of polycube splines and develops novel modeling techniques for using the polycube splines in solid modeling and shape computing. Polycube splines are essentially a novel variant of manifold splines which are built upon the polycube map, serving as its parametric domain. Our rationale for defining spline surfaces over polycubes is that polycubes have rectangular structures everywhere over their domains except a very small number of corner points. The boundary of polycubes can be naturally decomposed into a set of regular structures, which facilitate tensor-product surface definition, GPU-centric geometric computing, and image-based geometric processing. We develop algorithms to construct polycube maps, and show that the introduced polycube map naturally induces the affine structure with a finite number of extraordinary points. Besides its intrinsic rectangular structure, the polycube map may approximate any original scanned data-set with a very low geometric distortion, so our method for building polycube splines is both natural and necessary, as its parametric domain can mimic the geometry of modeled objects in a topologically correct and geometrically meaningful manner. We design a new data structure that facilitates the intuitive and rapid construction of polycube splines in this paper. We demonstrate the polycube splines with applications in surface reconstruction and shape computing.

Keywords: Polycube splines, manifold splines, polycube map, affine atlas, solid modeling, shape computing

1 Introduction and Motivation

Real-world physical prototypes are frequently 2-manifolds of complex geometry and arbitrary topology. With the rapid advancement of modern 3D scanning technologies, CAD-based digital prototypes are routinely acquired in forms of raw points and/or triangular meshes. In order to enable downstream product development processes (e.g., accurate shape analysis, finite element simulation, and e-manufacturing) in CAE environments, discrete data inputs must be converted into continuous, compact representations for scientific computing and engineering applications. In order to model an arbitrary manifold in 3D using conventional spline schemes, current approaches will segment the manifold to many smaller open patches, cover each patch by a single coordinate system, so that each patch can be modelled by a spline surface. Finally, any generic approach must glue all the spline patches together by adjusting the control points and the knots along their common boundaries in order to ensure continuity of certain degree. Even worse, the entire segmenting and patching process is primarily performed manually, and it requires users' knowledge and skills, and for non-trivial topology and complicated geometry this is laborious and error-prone.

To overcome the above modeling difficulties and address the topological issue, we seek novel modeling techniques that would allow designers to directly define continuous spline models over any manifolds (serving as parametric domains). Such a global approach would have many modeling benefits, including no need of the transition from local patch definition to global surface construction via gluing, the elimination of non-intuitive segmentation and patching process, and ensuring the high-order continuity requirements. More

*Stony Brook University, e-mail: wanghy@cs.sunysb.edu
†Nanyang Technological University, e-mail: yhe@ntu.edu.sg
‡Stony Brook University, e-mail: xinli@cs.sunysb.edu
§Stony Brook University, e-mail: gu@cs.sunysb.edu
¶Stony Brook University, e-mail: qin@cs.sunysb.edu

importantly, we can expect a true "one-piece" representation for shapes of complicated topology, with a hope to automate the entire reverse engineering process (by converting points and/or polygonal meshes to spline surfaces with high accuracy) without human intervention. This paper presents polycube splines that serve for this need.

Towards the aforementioned goal, most recently the manifold splines proposed by Gu, He, and Qin [Gu et al. 2005] also aim to provide a technical solution for directly defining continuous surfaces over arbitrary manifold domains. In their work, a manifold can be equivalently treated as a set of coordinate charts in $\mathbb{R}^2$ via local parameterization, and these local charts are then glued coherently to form a complete manifold surface. As a result, manifold splines are essentially piecewise polynomials or rational polynomials defined on affine manifolds, whose transition functions between different charts are all affine transformation. Thus, the evaluation algorithms and other computational procedures are both efficient and robust. They have also showed that any planar spline schemes (defined over an open planar domain) which satisfy the parametric affine invariant property can be straightforwardly extended to manifolds of arbitrary topology within the manifold spline framework [He et al. 2006a; He et al. 2005; He et al. 2006b].

Despite this earlier success, certain drawbacks of manifold splines still remain and demand more powerful modeling techniques. First of all, there must be singularities for any closed manifold except tori. Hence, for a closed manifold of $g > 1$, there has to be singularities of the atlas which can not be covered by any chart within its collection set. The existence of singularities comes from the topological obstruction, which can not be avoided within the current manifold spline framework. Given a closed domain manifold of genus g, [Gu et al. 2005] proposed a method to compute the affine structure with Euler number $|2-2g|$ extraordinary points and showed that the induced transition functions are simply the translation. Although in theory singularity points are simply points without occupying any regions or areas, in practice "small" holes must be punched in order to enable the easy construction of manifold splines. Their earlier work makes no efforts to actually fill the "small" holes in the vicinity of extraordinary points, in spite of their theoretic contributions. In addition, given the fact that the number of singularities is actually fixed, but their positions are somehow globally related, which are determined by the intrinsic conformal structure of the underlying surface and are usually difficult to control, i.e., it is impossible to specify the locations of all the singularities on the domain manifold.

In this paper, we forge ahead with our new research efforts by developing the polycube splines, with a goal to further improve the current state of knowledge for manifold splines. In a nutshell, our polycube splines can be considered as a novel variant of manifold splines with many new and attractive modeling properties. Unlike the previous manifold splines, the polycube splines are built directly upon the polycube map, serving as its parametric domain. Because of its regularity, the polycube is now only covered by charts which are uniquely associated with faces and edges belonging to one of the cubes. As a result of the polycube map, all the corner points are now becoming singular. The key motivation for us to pursue the definition and construction of polycube splines is the fact that the polycube map offers a rectangular structure which for sure will facilitate geometric computing and shape analysis. Another main advantage of the polycube spline is that its parametric domain can mimic the geometry of any modeled objects in a topologically correct way, hence, it is much easier to isolate and control the position of the singularities. Furthermore, there are only two kinds of connectivity on the singularities, either valence 3 or valence 5, which can greatly simplify our procedures to handle extraordinary points. The polycube domain can be constructed to approximate the modeled geometry with better accuracy, but at the expense of more cubes and more charts. So, users will have freedom to control the complexity of the underlying parametric domain and place singularity points with great flexibility. Figure 1 demonstrates an example of our polycube splines. Similar to manifold splines, polycube splines also afford a general theoretic and engineering framework in which all the existing planar splines can be generalized to any polycube domain via affine structure. In this paper, we develop algorithms to construct T-splines over polycubes and demonstrate their applications in shape modeling and reverse engineering in order to take advantage of the properties of partition-of-unity, level-of-detail control, and hierarchical representation. It may be noted that other powerful spline schemes, such as triangular *B*-splines, can be employed in a similar fashion.

1.1 Contributions

The specific contributions of this paper are as follows:

1. We present a systematic way to construct polycube maps for surfaces of arbitrary topology. Our method is fundamentally different from Tarini *et al.*'s technique [Tarini et al. 2004] in that we do not need to compute the projection of the points from the 3D shape to the polycube, thus, the polycube can be flexibly constructed at any resolution and complexity.

2. We show that the introduced polycube maps naturally induce the affine structure by removing a finite number of corner points. Thus, polycube splines become a novel variant of manifold splines with many new and attractive properties (outlined above). Taking advantage of the low area distortion between the domain manifold and the smooth spline surface (because polycubes can be built to approximate the modeled geometry within any user-specified accuracy), the polycube splines can be constructed easily and robustly by using simple and regular charts and isolating all the user-controllable singularity points.

3. Polycube splines offer a general framework in which any existing planar spline scheme can be generalized to a polycube domain via affine structure. Especially, in this paper, we construct T-splines on polycubes and demonstrate the efficiency of polycube splines to model surfaces with high fidelity, while retaining the attractive properties of partition-of-unity, level-of-detail control, and hierarchical representation.

The remainder of this paper is organized as follows. We review the related work on splines and parametrizations in Section 2. We present the details of our algorithm to construct the polycube map of arbitrary topology in Section 3. We then discuss our construction algorithms for polycube splines and document experimental results with statistics and performance data in Section 4. Finally, we conclude our paper in Section 5 with future research directions.

2 Related Work and Background

This section briefly reviews some related work on T-splines, manifold splines, surface parameterization and polycube maps, in order to build up necessary background knowledge for us to develop polycube splines.

T-splines. In [Sederberg et al. 2003], Sederberg *et al.* presented the T-spline, a generalization of the non-uniform B-spline surfaces. Unlike tensor-product splines, T-spline control grids are no longer required to be totally regular. In particular, they permit T-junctions, and iso-parametric curves of control points need not to traverse the entire column/row of control grids. Therefore, T-splines enable a true local refinement without introducing additional, unnecessary control points in nearby regions. Sederberg *et al.* also developed

an algorithm to convert industry standard NURBS surfaces into T-spline surfaces, in which a large percentage of superfluous control points are eliminated [Sederberg et al. 2004]. Recently, Li *et al.* presented an automatic technique to convert polygonal meshes to T-splines using periodic global parameterization [Li et al. 2006]. Zheng *et al.* developed techniques for adaptively fitting T-splines to functional data [Zheng et al. 2005]. Wang and Zheng addressed the issue of control point removal for T-spline surfaces [Wang and Zheng 2006].

Manifold splines. Gu *et al.* [Gu et al. 2005] developed a general theoretical framework of manifold splines in which spline surfaces defined over planar domains can be systematically generalized to any manifold domain of arbitrary topology (with or without boundaries). He *et al.* further developed modeling techniques for applications of manifold splines using triangular *B*-splines [He et al. 2006a], Powell-Sabin splines [He et al. 2005] and T-splines [He et al. 2006b].

Surface parameterization and polycube maps. Surface parameterization has been a very active research area in the past decade [Floater and Hormann 2005]. Parameterization can be viewed as a mapping from 3D surface to a canonical domain in 2D. Since isometric mappings only exist in very special cases, many approaches to surface Euclidean parameterization therefore attempt to find a mapping which is either conformal (i.e., no angular distortion) [Sheffer et al. 2005; Gu and Yau 2003; Lévy et al. 2002; Ray et al. 2006; Jin et al. 2004; Kharevych et al. 2006], or equiareal (i.e., no area distortion) [Maillot et al. 1993; Surazhsky and Gotsman 2003]. Hyperbolic parametrization for high genus number surfaces is presented in [Jin et al. 2006]. Spherical parametrization for genus zero surfaces are introduced in [Gotsman et al. 2003; Gu et al. 2004]. Tarini *et al.* pioneered the concept of Polycube maps in [Tarini et al. 2004], which minimizes both the angular distortion and area distortion. They demonstrated that polycube maps naturally lead to a seamless texture mapping method that is simple enough to be implemented in currently available graphics hardware.

3 Construction of Polycube Maps

In this section, we explain in details our algorithm for constructing polycube maps for surfaces of arbitrary topology. The key difference between the techniques employed in [Tarini et al. 2004] and ours in this paper is that Tarini *et al.*'s technique is trying to find the one-to-one mapping of the 3D shape and polycube extrinsically, which typically requires the projection of points from one shape to the other. As a result, their method is usually quite difficult to handle cases where the two shapes differ too much and the point projection does not establish the one-to-one correspondence. In contrast, our method aims to compute such a mapping in an intrinsic way. We first conformally map the 3D shape and the polycube to the same canonical domains (e.g., sphere, Euclidean plane, or hyperbolic disk), then we construct a map between these two domains, which induces a one-to-one map between the 3D shape and the polycube. Since our method avoids the direct projection of the 3D shape to the polycube, the polycube can be constructed independent of the actual geometry of 3D shape, allowing different complexity and resolution for the polycube.

For the generality of our algorithm, we assume that all surfaces acquired from scanning devices are represented as triangular meshes, which is a simplicial complex embedded in $\mathbb{R}^3$. Constructing the polycube map is equivalent to seeking a bijective map between the 3D model and the polycube. Our method for establishing such a mapping varies according to different topologies of surfaces: genus zero surfaces, genus one surfaces, and surfaces of high genus.

3.1 Riemann Uniformization Metric

Suppose a surface S is embedded in $\mathbb{R}^3$, then it has a Riemannian metric, which is represented by its first fundamental form, induced from the Euclidean metric of $\mathbb{R}^3$, denoted by $\mathbf{g}$. Suppose $u : S \to \mathbb{R}$ is a scalar function defined on S, then it can be verified that $e^{2u}\mathbf{g}$ is another Riemannian metric on S, denoted by $\bar{\mathbf{g}}$. It can be proven that angles measured by $\mathbf{g}$ are equal to those measured by $\bar{\mathbf{g}}$. Therefore, $\bar{\mathbf{g}}$ is conformal to $\mathbf{g}$ and now e^{2u} is called the conformal factor.

Riemann uniformization theorem [Jost and Simha 1997] states that for any surface S, there exists a unique conformal metric, such that it induces constant Gaussian curvature K and zero geodesic curvature.

$$K = \begin{cases} +1, & \chi(S) > 0 \\ 0, & \chi(S) = 0 \\ -1, & \chi(S) < 0 \end{cases}$$

Such kind of metric is called the uniformization metric of S.

We compute the uniformization metric with heat flow method [Gu et al. 2004] for genus zero surfaces, holomorphic 1-form method [Gu and Yau 2003; Jin et al. 2004] for genus one surfaces, and hyperbolic ricci flow method [Jin et al. 2006] for surfaces with genus greater than one.

In the following, we use notations M and P to denote the 3D model and its polycube approximation (serving as the parametric domain), respectively.

The overall flow of our algorithm for establishing the one-to-one mapping can be summarized as follows:

1. Given a 3D model M from data acquisition, construct a polycube P which roughly resembles the geometry of M and is of the same topology of M.
2. Compute the uniformization metric of M and embed M in the canonical domain D_M, which is a domain in $\mathbb{S}^2$, $\mathbb{E}^2$ or $\mathbb{H}^2$, i.e., $\phi_M : M \to D_M$.
3. Compute the uniformization metric of P and embed P in the canonical domain D_P, i.e., $\phi_P : P \to D_P$.
4. Construct the map $\phi_{D_M \to D_P} : D_M \to D_P$.
5. Finally, the composition $\phi_{M\to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M$ gives the desired polycube map from M to P as shown in equation (1).

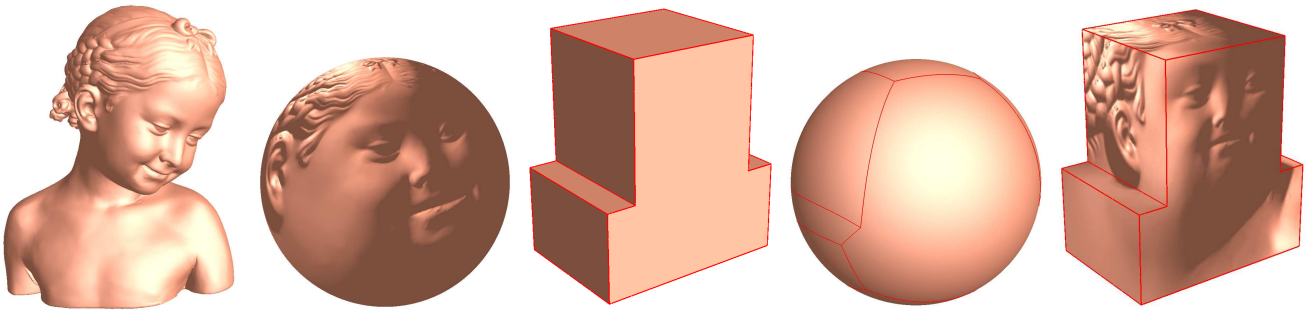
$$\begin{array}{ccc} M & \xrightarrow{\phi_{M\to P}} & P \\ \downarrow{\scriptstyle \phi_M} & & \downarrow{\scriptstyle \phi_P} \\ D_M & \xrightarrow[\phi_{D_M \to D_P}]{} & D_P \end{array} \tag{1}$$

Note that, our construction method varies depending on different types of surfaces. Genus zero surfaces are mapped to the unit sphere $\mathbb{S}^2$ with positive curvature $\bar{K} = 1$. Genus one surfaces are mapped to Euclidean plane $\mathbb{E}^2$ with zero curvature $\bar{K} = 0$. Surfaces of high genus are mapped to hyperbolic disk $\mathbb{H}^2$ with negative curvature $\bar{K} = -1$.

3.2 Harmonic Map

Definition (Harmonic Map). Given a Riemann surface M, a map $\phi : M \to \mathbb{R}^3$ is harmonic, if

$$\Delta\phi \equiv 0,$$



(a) M (b) $\phi_M : M \to \mathbb{S}^2$ (c) P (d) $\phi_P : P \to \mathbb{S}^2$ (e) $\phi : M \to P$

Figure 2: Conformal mapping a genus zero surface to the unit sphere induces the genus zero conformal polycube map. The genus zero Bima model M in (a) is conformally mapped to the unit sphere in (b). The polycube P in (c) is also mapped to the sphere in (d). By aligning three feature points on M and P, we can uniquely determine the conformal polycube map $\phi : M \to P$ between M and P.

where Δ is the Laplace-Beltrami operator of the surface M.

Suppose M is a genus zero closed surface, and $\phi : M \to \mathbb{S}^2$ is a harmonic map from the surface to the unit sphere, then ϕ is also conformal.

The following heat flow method can deform a map to be harmonic

$$\frac{d\phi}{dt} = -\Delta\phi.$$

In practice, the initial map ϕ is selected to be the Gauss map. The conformal map is not unique, two such maps differ by a Möbius transformation of the unit sphere. For more details, we refer the readers to [Gu et al. 2004].

3.3 Holomorphic 1-form

Definition (Holomorphic 1-form). Given a Riemann surface M with a conformal structure $\mathfrak{A}$, a holomorphic 1-form ω is a complex differential form, such that on each local chart $(U,\phi) \in \mathfrak{A}$

$$\omega = f(z)dz$$

where $f(z)$ is an analytic function, $z = u + iv$ is the local parameter in the complex form.

Genus zero surface has no holomorphic 1-forms. The holomorphic 1-forms of closed genus g surface form a g complex dimensional linear space, denoted as $\Omega(M)$. A conformal atlas can be constructed by using a basis of $\Omega(M)$. This is the method derived in [Gu and Yau 2003; Jin et al. 2004]. Considering its geometric intuition, a holomorphic 1-form can be visualized as two vector fields $\omega = (\omega_x, \omega_y)$, such that the curl of ω_x and ω_y equal zeros. Furthermore, one can rotate ω_x about the normal a right angle to arrive at ω_y,

$$\bigtriangledown \times \omega_x = 0, \bigtriangledown \times \omega_y = 0, \omega_y = n \times \omega_x.$$

By integrating a holomorphic 1-form, an affine atlas can be easily constructed. The local parameters are obtained by integrating the 1-form on the surface (see [Gu and Yau 2003; Jin et al. 2004] for the details).

3.4 Hyperbolic Ricci Flow

Hyperbolic Ricci flow is explained in details in [Jin et al. 2006]. A circle packing on a mesh is to associate a circle with each vertex, circles intersect each other. A mesh with circle packing is denoted as (M,Γ,Φ), where M represents the triangulation (connectivity) with vertex set V, edge set E and face set F, $\Gamma = \{\gamma_i, v_i \in V\}$ are the vertex radii and $\Phi = \{\Phi_{ij}, e_{ij} \in E\}$ are the angles associated with each edge. A circle packing (M,Φ,Γ) uniquely determines a so called *circle packing metric*. The length l_{ij} associated with the edge e_{ij} is computed using the hyperbolic cosine law. A *discrete conformal mapping* $\tau : (M,\Gamma,\Phi) \to (M,\bar{\Gamma},\Phi)$ solely changes the vertex radii Γ, but preserves the intersection angles Φ.

The discrete Gaussian curvature K_i at an interior vertex v_i with surrounding face f_{ijk} is defined as

$$K_i = 2\pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, \qquad v_i \notin \partial M, \tag{2}$$

where θ_i^{jk} is the corner angle of f_{ijk} at v_i. While the discrete Gaussian curvature for a boundary vertex v_i is defined as

$$K_i = \pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, \qquad v_i \in \partial M. \tag{3}$$

Given a mesh with circle packing metric (M,Γ,Φ), the discrete hyperbolic Ricci flow is defined as

$$\frac{\partial \gamma_i}{\partial t} = -\sinh\gamma_i K_i \tag{4}$$

A solution to equation (4) exists and is *convergent*

$$\lim_{t\to\infty} K_i(t) = 0.$$

In theory, the discrete Ricci Flow is guaranteed to be exponentially convergent [Chow and Luo 2003] [Jin et al. 2006].

3.5 Genus-zero Polycube Map

Genus zero surfaces are topologically equivalent to sphere. Thus, we use sphere as the canonical domain for both M and P. A method based on non-linear heat flow to construct conformal maps between a closed genus zero surface and the unit sphere $\mathbb{S}^2$ is introduced in [Gu et al. 2004].

After computing the maps $\phi_M : M \to \mathbb{S}^2$ and $\phi_P : P \to \mathbb{S}^2$, we need to find a map $\phi_{D_M \to D_P} : \mathbb{S}^2 \to \mathbb{S}^2$ which can align their major features. For example, we want to align the eyes and nose of the Bimba model (see Figure 2) to be at certain positions on the polycube. To

do so, we conformally map the sphere to the plane using stereographic projection

$$\tau : (x,y,z) \to (\frac{2x}{1-z}, \frac{2y}{1-z}),\ (x,y,z) \in \mathbb{S}^2. \tag{5}$$

We then use a special conformal map from the plane to itself, a Möbius transformation, to move three arbitrary feature points into any new desired positions. Suppose for the first surface, the three feature points are z_0, z_1 and z_2. We first construct the Möbius transformation which takes them into 0, 1, and ∞:

$$\psi_1 = \frac{(z-z_0)(z_1-z_2)}{(z-z_2)(z_1-z_0)}.$$

We then construct ψ_2 for three positions on P in a similar way. Then $\psi_1^{-1} \circ \psi_2$ maps the feature points on the second surface into those on the first one. Finally, the conformal map $\phi_{D_M \to D_P} : \mathbb{S}^2 \to \mathbb{S}^2$ is defined as

$$\phi_{D_M \to D_P} = \tau^{-1} \circ \psi_2^{-1} \circ \psi_1 \circ \tau.$$

Note that the polycube map $\phi_{M \to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M$ is conformal since each sub-map is conformal.[1]

Figure 2 demonstrates the construction pipeline of defining conformal polycube map for genus zero surfaces.

3.6 Genus-one Polycube Map

Suppose M is a genus one closed surface, ω is a holomorphic 1-form. By integrating ω, M can be periodically mapped to the plane, each period is called a fundamental polygon. Each canonical fundamental polygon of genus one surface is a parallelogram. For details, we refer readers to the work [Gu and Yau 2003].

Given arbitrary two parallelograms, there exists a unique affine map to map one to the other, such that corners are mapped to corners, sides are mapped to sides.

The fundamental polygons of M and P, D_M and D_P, are parallelograms. Denote the unique affine map between them as $\phi_{D_M \to D_P}$, then the polycube map $\phi_{M \to P} : M \to P$ is formulated as

$$\phi_{M \to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M.$$

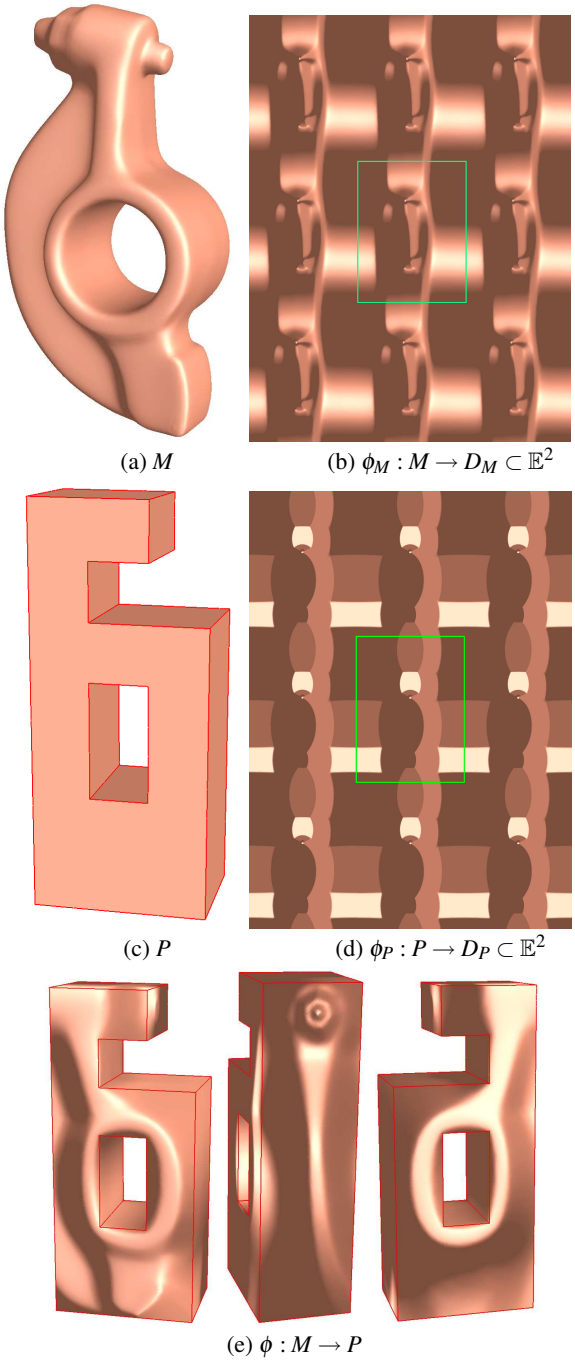
Figure 3 demonstrates the above mapping method for constructing a polycube map of the Rockerarm model. The polycube mesh is manually built. Then both the Rockerarm mesh and the polycube model are parameterized using the holomorphic 1-form method. Their fundamental polygons are extracted and mapped by an affine map. The affine map further induces a bijective map between the Rockerarm model and the polycube.

3.7 High Genus Polycube Map

Given a high genus surface with simple geometry like the 3-hole torus model shown in figure 10, the polycube map can be constructed using the techniques in [Tarini et al. 2004]. However, for surfaces with complicated geometries like the model in figure 5, the direction projection techniques in $\mathbb{R}^3$ hardly generate bijective maps. We use hyperbolic parameterization method instead.

Given a surface M with genus $g \geq 1$, a set of $2g$ curves $\{a_1, b_1, a_2, b_2, \ldots, a_g, b_g\}$ are called a canonical homology basis if they meets the following criteria:

[1] Strictly speaking, the map $\phi_P : P \to \mathbb{S}^2$ is conformal everywhere except at the corners of the polycube.



(a) M (b) $\phi_M : M \to D_M \subset \mathbb{E}^2$

(c) P (d) $\phi_P : P \to D_P \subset \mathbb{E}^2$

(e) $\phi : M \to P$

Figure 3: Euclidean structure induces the genus-one polycube map. The genus one Rockerarm model M in (a) is conformally mapped to the Euclidean plane in (b). The fundamental domain is a rectangle region enclosed by the green boundary in (b). Then, a polycube P in (c) is also parameterized over the rectangular region in the same way in (d). By matching the two fundamental regions in (b) and (d) via an affine map, the conformal polycube map for the Rockerarm model is established.

1. All the curves meet at a single base point, v.
2. Each pair of curves $\{a_i, b_i\}$ algebraically intersect each other exactly once.
3. No curve in another pair a_j, b_j algebraically intersects either of a_i, b_i.

With the uniformization metric, M with $g > 1$ can be periodically mapped onto the hyperbolic space $\mathbb{H}^2$. We use the Poincaré hyperbolic disk model to represent the hyperbolic space $\mathbb{H}^2$. The Poincaré disk is a two-dimensional space defined in the unit disk $\{z \in \mathbb{C} : |z| < 1\}$ on the complex plane $\mathbb{C}$ with hyperbolic metric. The hyperbolic metric is defined as

$$ds^2 = \frac{dzd\bar{z}}{(1-\bar{z}z)^2}.$$

The geodesic (hyperbolic lines) in the Poincaré disk are Euclidean circular arcs perpendicular to the boundary $|z| = 1$. The rigid motions in the hyperbolic plane are the Möbius transformations $z \to w$, $z \in \mathbb{C}$ with the form

$$w = e^{i\theta}\frac{z - z_0}{1 - \bar{z}_0}, \tag{6}$$

where z_0 is an arbitrary point inside the unit disk.

The canonical homology basis are mapped to geodesics. Each period is a polygon and called a fundamental polygon. Then, the fundamental domain of M is a polygon with $4g$ sides, corresponding to $\{a_i, b_i, a_i^{-1}, b_i^{-1}\}_{i=1}^{g}$. Figure 4 illustrates the canonical homology basis and the hyperbolic embedding with the uniformization metric for a genus 2 model.

In order to find the map between M and P, we compute their hyperbolic parameterizations by solving the discrete hyperbolic Ricci flow in (4). The readers are referred to [Jin et al. 2006] for more details. Then, similar to the genus zero case, a harmonic map $\phi_{D_M \to D_P}$ is constructed such that maps the fundamental polygon of M to the fundamental polygon of P. Finally, the polycube map is constructed as

$$\phi_{M \to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M.$$

Figure 5 demonstrates the example of polycube map for a genus-3 surface and highlight our construction pipeline.

4 Polycube Spline Definition and Algorithm

After mapping the original model onto its polycube approximation using the above theory and method, it now sets the stage to define polycube splines and actually build up polycube splines using the polycube map. This will enable the automatic reverse engineering from polygonal models initially acquired to a more compact spline representation with high accuracy.

4.1 The Affine Atlas via Polycube Map

We construct an affine atlas from the polycube map. Each face and edge on the polycube are associated with its own local chart. Each face chart covers only interior points of corresponding face and leaves off all the edges of the face. Each edge chart covers interior points of the edge but leaves off corner vertices. Furthermore, there are overlaps between face charts and edge charts. The transition functions between overlapped edge and face charts are simply translations and rotations. Note that there is **NO** vertex chart for the corner vertex, i.e., the corners are singular points. Therefore, by removing all the corners, polycube map naturally induces the affine

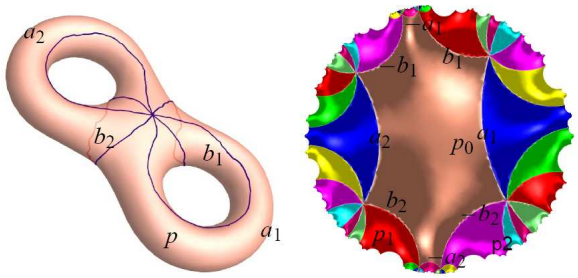


Figure 4: A genus two surface with a set of canonical fundamental group generators $\{a_1, b_1, a_2, b_2\}$ is shown on the left. A finite portion of its universal covering space is shown on the right. Different fundamental domains are drawn in different colors. The boundary of each fundamental domain is the preimage of $a_1 b_1 a_1^{-1} b_1^{-1} a_2 b_2 a_2^{-1} b_2^{-1}$. The points $\{p_0, p_1, p_2\}$ are the primages of p on the surface.

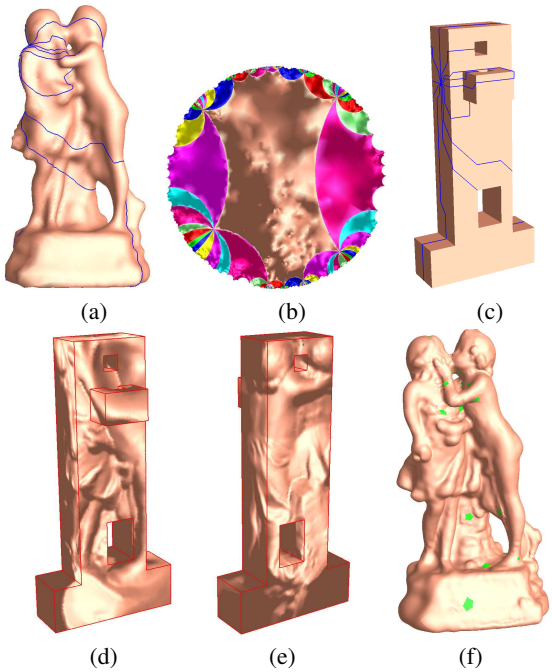
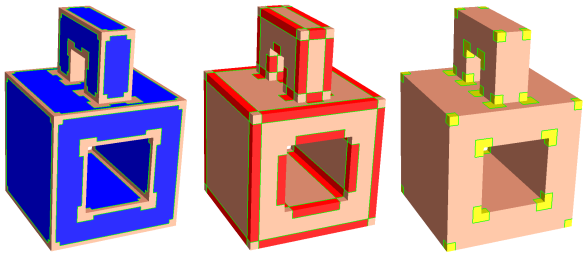


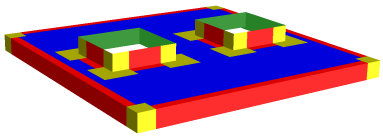
Figure 5: Hyperbolic structures induce the high genus polycube map. The canonical homology basis of the genus-3 sculpture model are colored in blue in (a). (b) shows the isometric embedding of its universal covering space on the Poincaré hyperbolic disk. We compute the hyperbolic uniformization metric of the polycube in (c) using a similar approach. The canonical homology basis are drawn in blue in (c). By establishing the correspondence between the fundamental domains, we construct the polycube map (shown in (d) and (e)) between (a) and (c). (f) shows the Polycube spline surface obtained via affine structure induced by the polycube map. The extraordinary points are colored in green.

structure. Figure 6 highlights face and edge charts of a polycube. The extraordinary points are colored in yellow.

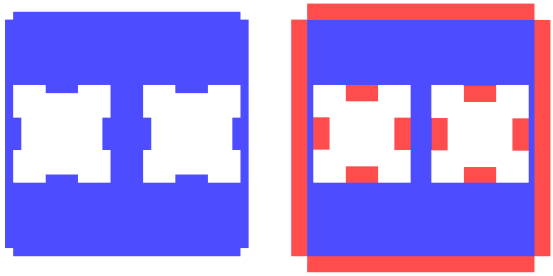
In [Gu et al. 2005], they have pointed out that any planar spline schemes which satisfy the parametric affine invariant property can be generalized to manifold domain via affine structure. By removing all the corner points, a polycube domain is just an affine manifold preserving the affine structure. Therefore, we can define spline surfaces on polycube directly.



(a) Face charts (b) Edge charts (c) Singularities



(d) Close-up of one face chart and its associated edge charts and singularities



(d) Face chart (e) Transition function

Figure 6: Polycube map induces affine structure. The polycube is covered by face and edge charts. Each face chart (drawn in blue) covers only interior points of corresponding face and leaves off all the edges of the face. Each edge chart (drawn in red) covers interior points of the edges but leaves off corner vertices. The corners (drawn in yellow) are singularities which are NOT covered by any charts. We highlight one face chart and its associated edge charts and singularities in (d). By flattening the edge charts, we get the planar domain shown in (e). Note that the transition functions between overlapped edge and face charts are simply translations and rotations. Therefore, by removing all the corners, the open polycube $P \setminus C$ has the affine structure.

4.2 Hierarchical Spline over Polycube Map

Denote the corners of P by C. Given the polycube map P covered by a set of edge and face charts (U_i, ϕ_i), the polycube spline can be formally defined as follows:

$$\mathbf{F}(\mathbf{u}) = \sum_i \mathbf{F}_i(\mathbf{u}),\ \mathbf{u} \in P \setminus C, \tag{7}$$

where $\mathbf{F}_i = \sum_j \mathbf{c}_j B_j(\phi_i(\mathbf{u}))$ is the spline surface defined on (edge or face) chart (U_i, ϕ_i) and B_j is the basis function.

Note that Equation (7) is a general representation. Any planar spline schemes which satisfy the parametric affine invariant property can be generalized to polycube domain using this formula. In this paper, we choose to use the powerful T-spline to illustrate our spline construction procedure. The key reason that we choose T-spline is that each face chart of the polycube is nothing more than a union of rectangles, conventional tensor-product splines are special cases of T-splines, and they are all naturally defined over rectangular regions. More importantly, the hierarchical definition and level-of-detail control are attractive features in practice. It may be noted that, though we make use of T-splines in our current implementation, our polycube spline framework enables other popular spline schemes.

Recall that for every control point in the T-mesh, the covering region of its basis function is a rectangle, whose side lengths (knot vectors) are determined by the connectivity of the T-mesh. In polycube T-splines, we follow the rules defined in [Sederberg et al. 2003; Sederberg et al. 2004]. We further require that on each chart, the basis functions vanishes outside the boundary of the chart. Thus, the face charts are totally separate from each other. Each edge chart connects two face charts (one face chart if it is a boundary edge and not shared by two faces). Therefore, given an arbitrary parameter $\mathbf{u} \in P \setminus C$, it may be covered by a single face chart, or a single edge chart, or by one face chart and one edge chart.

On each (edge and face) chart (U_i, ϕ_i), the spline patch is defined as a point-based spline whose control points form a T-mesh:

$$\mathbf{F}_i(\mathbf{u}) = \sum_j \mathbf{c}_j B_j(\phi(\mathbf{u})), \mathbf{u} \in U_i, \tag{8}$$

where $\mathbf{c}_j \in \mathbb{R}^3$ are the control points.

Given an arbitrary parameter $\mathbf{u} \in P \setminus C$, the spline evaluation can be carried out as follows:

1. Find the set of charts which cover this point $\mathbf{u}$. This set V contains one face chart, or one edge chart, or one face chart and one edge chart.
2. The function value is the partition of unity of the spline patches in the above chart(s), i.e.,

$$\mathbf{F}(\mathbf{u}) = \frac{\sum_{i \in V} \sum_j \mathbf{c}_j B_j(\phi_i(\mathbf{u}))}{\sum_{i \in V} \sum_j B_j(\phi_i(\mathbf{u}))}.$$

We now have to address the problem of how to find the best approximation of a given polygonal mesh M with vertices $\{\mathbf{p}\}_{k=1}^m$ using polycube splines. We resort to the techniques of data fitting via optimization.

We minimize a linear combination of interpolation and fairness functionals, i.e.,

$$\min E = E_{dist} + \lambda E_{fair}. \tag{9}$$

The first part is

$$E_{dist} = \sum_{k=1}^{m} \|\mathbf{F}(\mathbf{u}_k) - \mathbf{p}_k\|^2,$$

where $\mathbf{u}_k \in P \setminus C$ is the parameter for $\mathbf{p}_k$, $k = 1, \ldots, m$.
The second part E_{fair} in (9) is a smoothing term. A frequently-used example is the thin-plate energy,

$$E_{fair} = \iint_{P \setminus C} (\mathbf{F}_{uu}^2 + 2\mathbf{F}_{uv}^2 + \mathbf{F}_{vv}^2) du dv.$$

Note that both parts are quadratic functions of the unknown control points, leading to a linear system.

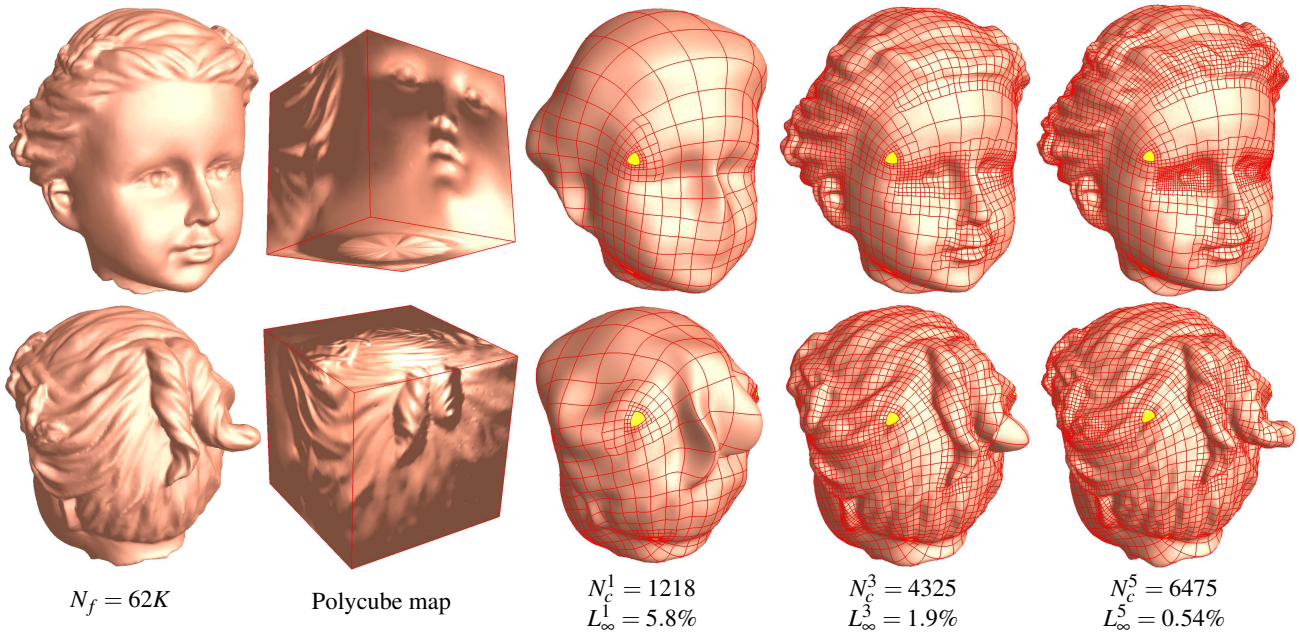


Figure 7: Hierarchical surface reconstruction. N_c^i and L_∞^i are the number of control points and maximal fitting error in iteration i. N_f is the number of faces in the polygonal model originally acquired. The input data is normalized to a unit cube in order to compute its fitting error.

We solve Equation (9) for unknown control points using Conjugate Gradient method. The value and gradient of the interpolation functional and fairness functional can be computed straightforwardly.

Our polycube T-splines defined using polycube maps are a new variant of general manifold T-splines with many more attractive properties. We shall employ the similar modeling strategy used in the hierarchical construction of manifold T-splines [He et al. 2006b]. Because of the existence of T-junctions, we will be able to control the approximation accuracy of the polycube T-splines locally by specifying the maximal fitting tolerance $L_\infty = \max \|\mathbf{F}(\mathbf{u}_i) - \mathbf{p}_i\|$, $i = 1, \ldots, m$. If the current surface does not satisfy this criterion, we employ adaptive refinement to introduce new degrees of freedom into the surface representation to improve the fitting quality. Because of the natural and elegant hierarchial structure of T-splines, this step can be done easily. Suppose a domain rectangle I violates the criterion and denote L_∞^I the L_∞ error on rectangle I. If the $L_\infty^I > 2\varepsilon$, split the rectangle I using 1-to-4 scheme; Otherwise, we divide I into two rectangles by splitting the longest edge. After adaptive refinement, we then re-calculate the control points until the maximal fitting tolerance is satisfied.

Figure 7 shows the entire procedure of hierarchical fitting of the Head model. The initial spline surface contains only 1218 control points and the maximal error $L_\infty = 5.8\%$. Through five iterations, we can obtain a much more refined spline surface with 6475 control points by inserting only necessary control points. The maximal fitting error reduces to 0.54%. As shown in the close-up view (Figure 8), our hierarchical data fitting procedure can produce high quality polycube T-splines with high-fidelity and we will be able to recover all the surface details.

4.3 Handling the Extraordinary Points

In [Gu et al. 2005], Gu *et al.* proved that manifold splines *MUST* have singularities if the domain manifold is closed and not a torus. They also proposed a method to compute the affine structure of a closed genus g manifold with Euler number of extraordinary points. However, they failed to develop any practical methods to actually filling those holes in the vicinity of singularity points. In this paper, because of the regularity of polycubes serving as parametric domains, we shall address the issue of hole-filling and present our method to handle extraordinary points.

The number of extraordinary points of polycube splines are determined by the geometry of the underlying polycube, i.e., each corner is a singularity. Note that the valence of a corner is either three or five, i.e., the holes are either 6 or 10-sided. We want to find a blending surface patch to fill the hole such that the surface is C^2 inside and G^1 along the boundaries of the holes. Naturally, there are many approaches available to fill a polygonal hole satisfying the above boundary condition.

In this paper, we want to use a spline surface to fill the singularities. In particular, we shall minimize the thin-plate energy of the hole-filling surfaces to improve the overall smoothness of the surface. Thus, our goal is to solve the following optimization problem:

$$E(s) = \iint_\Omega (\frac{\partial^2 s}{\partial u^2})^2 + 2(\frac{\partial^2 s}{\partial u \partial v})^2 + (\frac{\partial^2 s}{\partial v^2})^2 du dv. \qquad (10)$$

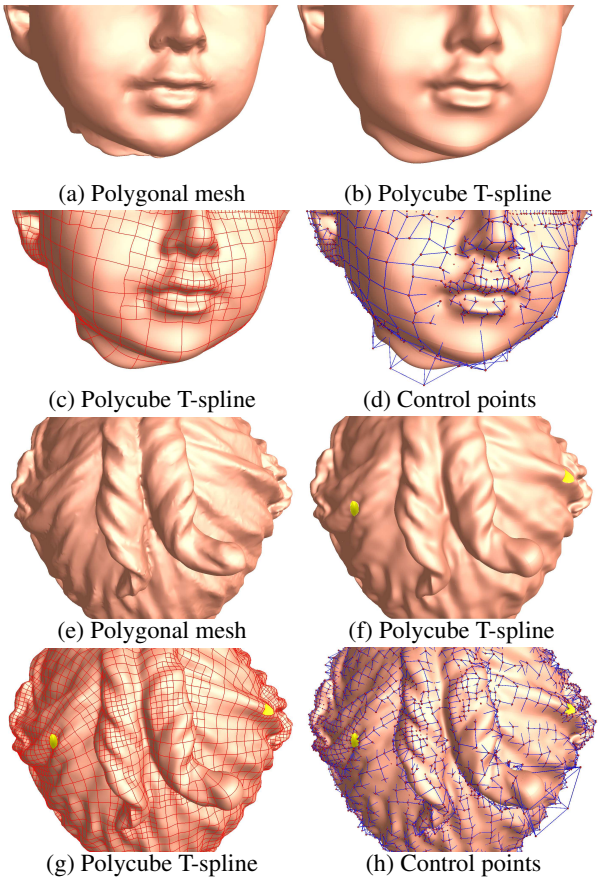
Our strategy to fill the hole is to find s^* solving the following minimization problem:

$$min\{E(s) : s|_{\partial\Omega} = f,\ \frac{\partial s}{\partial u} \times \frac{\partial s}{\partial v}|_{\partial\Omega} = n\}. \qquad (11)$$

where f and n are the boundary positions and normals.

In our current implementation, we use a cubic triangular B-spline to handle the hole-filling problem because of its flexibility in the domain construction and its potential to match with any number of sides of holes. The boundary conditions are represented by several sampling points on the boundary of the spline surface. The boundary position constraints naturally lead to a system of linear equations on the control points. The normal constraints are expressed as

$$< \frac{\partial s}{\partial u}, n >= 0, \ < \frac{\partial s}{\partial v}, n >= 0.$$



(a) Polygonal mesh (b) Polycube T-spline (c) Polycube T-spline (d) Control points (e) Polygonal mesh (f) Polycube T-spline (g) Polycube T-spline (h) Control points

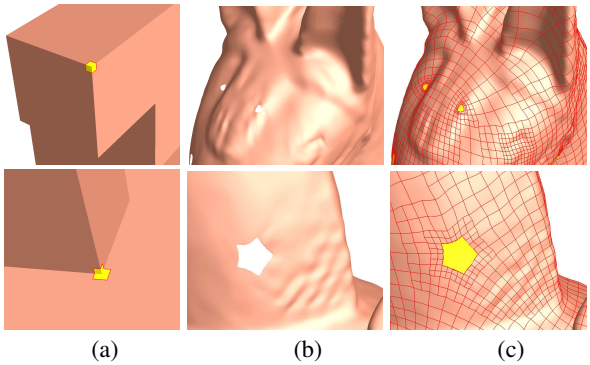
Figure 8: Close-up views of the reconstructed details. (a) and (e) show the original polygonal model. (b) and (f) show the polycube spline surfaces. (c) and (g) highlight the T-junctions on the spline surfaces. (d) and (h) show the spline overlaid with the control points.

Therefore, Equation (11) is a linear least-square problem with linear constraints, which can be solved easily using Lagrange Multiplier method. Figure 9 demonstrates the procedure pipeline to handle the extraordinary points.

4.4 Discussions

Polycube T-splines unify T-splines and manifold splines elegantly to form a new shape representation for surfaces of arbitrary topology. In this subsection, we compare polycube T-splines with the manifold T-spline proposed in [He et al. 2006b]

- The affine structure of manifold T-spline is induced from the conformal structure, which has $|2-2g|$ singularities for closed surface of genus g. The affine structure of polycube T-spline is also induced from the conformal structure. However, the number of singularities is determined by the geometry of the polycube, i.e., the number of corner points. Therefore, for a genus one surface, manifold T-splines are well defined everywhere. Polycube T-spline could have many singularities.
- In manifold T-spline, each vertex on the domain manifold and its two-ring neighbors form a chart. In the polycube T-spline, the number of charts is much less than that of manifold T-splines.
- In both spline schemes, the affine structure are induced from conformal structure. Hence, there is no angle distortion in each coordinate chart (U_i, ϕ_i). However, the area distortion in the mapping ϕ_i could be much bigger than that of the polycube T-splines. Therefore, polycube T-spline is much easier to construct.
- The valence of extraordinary points of manifold T-splines is eight, i.e., the hole is eight-sided. The valence of singularities of polycube T-spline is either three or five, thus, the hole is either six or ten-sided.



(a) (b) (c)

Figure 9: Handling singularities. Corners (shown in (a)) on the polycube are extraordinary points, which are corresponding to the holes on the polycube splines (shown in (b)). Note that the valence of extraordinary points are either 3 or 5, thus the holes are either 6 or 10-sided (shown in (b)). For each hole, we construct a cubic triangular B-spline surface which minimizes the thin-plate energy functional (11) and satisfies the boundary condition. (c) shows the final results after hole-filling. Holes are colored in yellow.

4.5 Experimental Results

Our prototype system is implemented in C++ on windows platform. We built a complete system for computing the polycube maps, affine structures and polycube T-splines. We tested our algorithms on various models from genus zero to genus three. The statistics of the test cases are shown in Table 1. Figure 7 illustrates the hierarchical surface reconstruction. As shown in Figure 7 and 8, we can get high-quality spline surfaces by gradually increasing the number of control points. More complicated models are shown in Figure 1 and 10. The results demonstrate both the theoretic rigor and feasibility in practice.

5 Conclusions

We have developed the polycube splines which not only inherit all the features of general manifold splines but also have new and more attractive properties of its own, including hierarchical representation, level-of-detail control, regular domain, partition-of-unity for basis functions, easy chart construction, and easy handling of extraordinary points. The polycube splines are naturally built upon the polycube map which serve as its parametric domain. The use of polycubes for spline surface definition and construction is the first attempt to take advantage of the rectangular structure over the boundary of polycubes, allowing the parametric domain to actually mimic the geometry of the modeled objects with lower area distortion while enforcing their topological consistence. We have presented our algorithms to construct polycube maps as the first step to enable spline construction over polycubes of arbitrary topology. We show that the introduced polycube maps easily induce the affine structures except at the finite number of corner points, where we

Table 1: Statistics of test examples. g, genus of polycube P; N_f, # of face charts; N_e, # of edge charts; N_s, # of singularities; N_c, # of control points; *rms*, root-mean-square error.

Object	g	N_f	N_e	N_s	N_c	*rms*
Head (Figure 7, 8)	0	6	12	8	6475	0.05%
Bimba (Figure 10)	0	10	24	16	10964	0.07%
Buddha (Figure 10)	0	10	24	16	11067	0.04%
Rockerarm (Figure 10)	1	14	38	24	4132	0.03%
3-hole Torus (Figure 10)	3	18	54	32	5180	0.02%
Isidore Horse (Figure 1)	3	12	30	20	12158	0.07%

also articulate our strategy for hole-filling. Through extensive experiments on various models, we demonstrate that polycube splines are a very good candidate for accurately representing complicated geometric models of arbitrarily complicated topology with low fitting errors and fewer control points (in comparison with polygonal models). Although the immediate application documented in this paper is reverse engineering, we foresee a broader application scope in solid modeling, shape analysis, data compression, FEM-based dynamic simulation, and virtual prototyping.

Acknowledgement

The Bimba, Isidore Horse, Buddha, Head models are provided courtesy of INRIA by the AIM@SHAPE Shape Repository.

References

CHOW, B., AND LUO, F. 2003. Combinatorial ricci flows on surfaces. *J. Differential Geometry*, 97–129.

FLOATER, M. S., AND HORMANN, K. 2005. Surface parameterization: a tutorial and survey. In *Advances in multiresolution for geometric modelling*, N. A. Dodgson, M. S. Floater, and M. A. Sabin, Eds. Springer Verlag, 157–186.

GOTSMAN, C., GU, X., AND SHEFFER, A. 2003. Fundamentals of spherical parameterization for 3d meshes. *ACM Trans. Graph. 22*, 3, 358–363.

GU, X., AND YAU, S.-T. 2003. Global conformal surface parameterization. In *Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing*, 127–137.

GU, X., WANG, Y., CHAN, T. F., THOMPSON, P. M., AND YAU, S.-T. 2004. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Transactions on Medical Imaging 23*, 8 (Aug.), 945–958.

GU, X., HE, Y., AND QIN, H. 2005. Manifold splines. In *ACM Symposium on Solid and Physical Modeling*, 27–38.

HE, Y., JIN, M., GU, X., AND QIN, H. 2005. A C^1 globally interpolatory spline of arbitrary topology. In *LNCS*, vol. 3752, 295–306.

HE, Y., GU, X.-F., AND QIN, H. 2006. Automatic shape control of triangular B-splines of arbitrary topology. *J. Comput. Sci. Technol. 21*, 2, 232–237.

HE, Y., WANG, K., WANG, H., GU, X., AND QIN, H. 2006. Manifold T-spline. In *Proceedings of Geometric Modeling and Processing*, 409–422.

JIN, M., WANG, Y., YAU, S.-T., AND GU, X. 2004. Optimal global conformal surface parameterization. In *IEEE Visualization*, 267–274.

JIN, M., LUO, F., AND GU, X. 2006. Computing surface hyperbolic structure and real projective structure. In *Symposium on Solid and Physical Modeling*, 105–116.

JOST, J., AND SIMHA, R. R. 1997. *Compact Riemann Surfaces: An Introduction to Contemporary Mathematics*. Springer-Verlag Telos.

KHAREVYCH, L., SPRINGBORN, B., AND SCHRÖDER, P. 2006. Discrete conformal mappings via circle patterns. *ACM Trans. Graph. 25*, 2, 412–438.

LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. 2002. Least squares conformal maps for automatic texture atlas generation. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 362–371.

LI, W.-C., RAY, N., AND LÉVY, B. 2006. Automatic and interactive mesh to T-spline conversion. In *EG/ACM Symposium on Geometry Processing*.

MAILLOT, J., YAHIA, H., AND VERROUST, A. 1993. Interactive texture mapping. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 27–34.

RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Transactions on Graphics*.

SEDERBERG, T. W., ZHENG, J., BAKENOV, A., AND NASRI, A. H. 2003. T-splines and T-NURCCs. *ACM Trans. Graph. 22*, 3, 477–484.

SEDERBERG, T. W., CARDON, D. L., FINNIGAN, G. T., NORTH, N. S., ZHENG, J., AND LYCHE, T. 2004. T-spline simplification and local refinement. *ACM Trans. Graph. 23*, 3, 276–283.

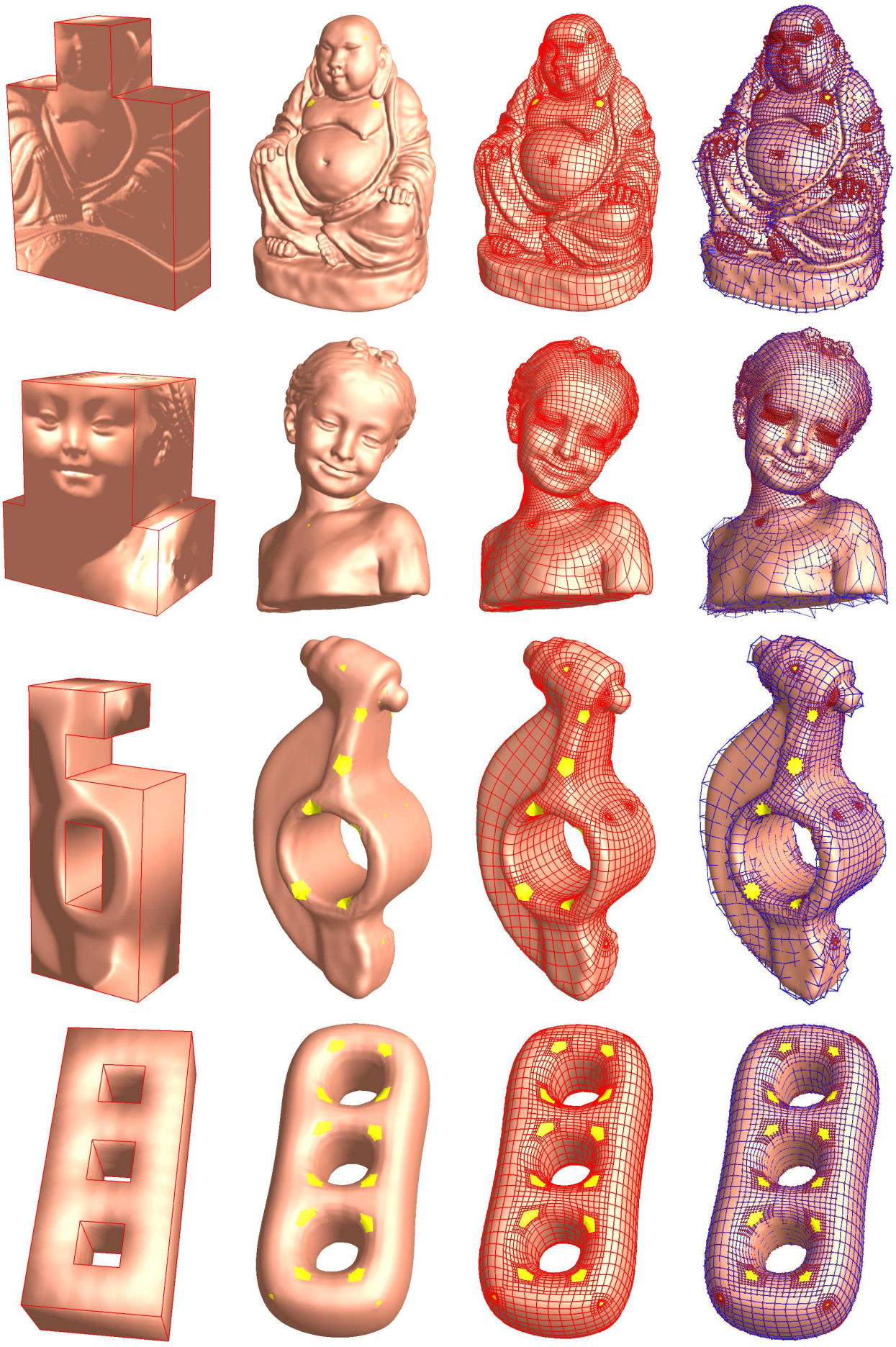
SHEFFER, A., LÉVY, B., MOGILNITSKY, M., AND BOGOMYAKOV, A. 2005. Abf++: fast and robust angle based flattening. *ACM Trans. Graph. 24*, 2, 311–330.

SURAZHSKY, V., AND GOTSMAN, C. 2003. Explicit surface remeshing. In *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 20–30.

TARINI, M., HORMANN, K., CIGNONI, P., AND MONTANI, C. 2004. Polycube-maps. *ACM Trans. Graph. 23*, 3, 853–860.

WANG, Y., AND ZHENG, J. 2006. Control point removal algorithm for T-spline surfaces. In *GMP*, 385–396.

ZHENG, J., WANG, Y., AND SEAH, H. S. 2005. Adaptive T-spline surface fitting to z-map models. In *GRAPHITE*, 405–411.



(a) Polycube map (b) Polycube T-spline (c) Polycube T-spline (d) Control points

Figure 10: Examples of Polycube T-splines.