Polycube Splines

Hongyu Wang [a], Ying He [b,*], Xin Li [a], Xianfeng Gu [a], Hong Qin [a],

[a] *Computer Science Department, Stony Brook University, NY, USA.*

[b] *School of Computer Engineering, Nanyang Technological University, Singapore.*

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.

Key words: Manifold Splines, T-splines, Affine Structure, Conformal Structure, Holomorphic 1-form, Harmonic Map, Discrete Ricci Flow, Polycube Map
PACS: Computational Geometry, Object Modeling, Geometric Algorithms

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 geometric design and 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. 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 task is laborious and error-prone.

To overcome the above modeling and design 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 and abutting, the elimination of non-intuitive segmentation and patching process, and ensuring the high-order continuity requirements. More importantly, we can expect a true "one-piece" representation for shapes of complicated topology, with a hope to au-

* Corresponding author. Address: 50 Nanyang Avenue, Blk N4, Nanyang Technological University, Singapore, 639798. Tel: +65 65141008; Fax: +65 67926559.

Email addresses: wanghy@cs.sunysb.edu (Hongyu Wang), yhe@ntu.edu.sg (Ying He), xinli@cs.sunysb.edu (Xin Li), gu@cs.sunysb.edu (Xianfeng Gu), qin@cs.sunysb.edu (Hong Qin).

tomate the entire reverse engineering process (by converting points and/or polygonal meshes to spline surfaces with high accuracy) without human intervention.

Towards the aforementioned goal, most recently the manifold splines proposed by Gu, He, and Qin [10] 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 [14,13,15].

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, [10] 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 in the finite dimension space. 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.

Aside from splines, subdivision surfaces have also been extensively investigated during the recent past for the continuous representation of discrete data inputs. It defines a smooth surface as the limit of a sequence of successive refinements from a given coarse polygonal mesh. All the chart transition functions are rotation, translation and scaling. Despite their modeling advantages for arbitrarily complicated geometry and topology, subdivision surfaces have two drawbacks: (1) accurate surface evaluation is frequently conducted via explicit, recursive subdivision since most subdivision schemes (especially those interpolatory schemes) do not allow closed-form analytic formulation for their basis functions; (2) extraordinary points depend on the connectivity of the control mesh and need special care, as their behaviors and smoothness properties differ significantly from other regular regions nearby. Subdivision surfaces can be considered special cases of manifold splines according to [10].

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 four kinds of connectivity on the singularities, valence 3 to 6, 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:

(i) 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 [29] 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.

(ii) 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

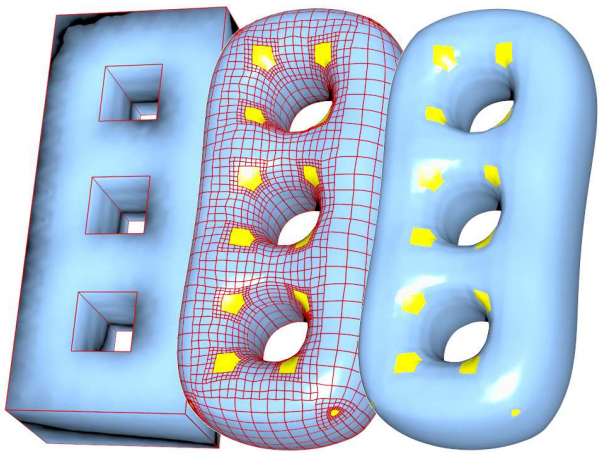


Fig. 1. T-splines on polycubes. The polycube serves as the parametric domain which mimics the geometry of the 3D model. All the corners are singularities which are colored in yellow.

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.

(iii) 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 parameterizations in Section 2. We present the detailed algorithms for constructing the polycube map in Section 3. Next, we show the hierarchical surface reconstruction in Section 4. Experimental results with statistics and performance data are also shown in Section 4. Finally, we conclude our paper in Section 5 with future research directions.

2. Related Work and Background

2.1. *T-splines*

In [25], Sederberg *et al.* pioneered 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 [26]. Zheng *et al.* developed techniques for adaptively fitting T-splines to functional data [34]. Wang and Zheng addressed the issue of control point removal for T-spline surfaces [30]. Yang *et al.* developed T-spline level sets for image segmentation and meshing non-uniform sampled and incomplete data [31,32]. Deng *et al.* introduced the polynomial spline functions over T-meshes, an extension of T-splines such that the splines are piecewise polynomials instead of rational functions [4]. Recently, Li *et al.* presented an automatic technique to convert polygonal meshes to T-splines using periodic global parameterization [22,24]. Li *et al.*'s method can be also viewed as manifold splines since the transition functions of the periodic global parameterization are composition of translations and rotations [24].

2.2. *Manifold Construction*

In essence, manifold construction is to model surfaces using charts. The shape (2-manifold) is covered by several charts. One builds functions on each chart. Due to certain continuity requirement of the transition functions between overlapping charts, the smoothness properties of the manifold functions are *automatically* guaranteed. Therefore, there are no restrictions/constraints on the control points. All the control points are free variables in the entire modeling process. Furthermore, manifold constructions can generate C^k smooth surfaces.

Grimm and Hugues [7] pioneered a generic method to extend *B*-splines to surfaces of arbitrary topology, based on the concept of overlapping charts. Cotrina *et al.* proposed a C^k construction on manifold [2,3]. Ying and Zorin [33] presented a manifold-based smooth surface construction method which has C^∞-continuous with explicit nonsingular parameterizations only in the vicinity of regions of interest.

Gu *et al.* [10] 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 [14] and Powell-Sabin splines [13].

2.3. *Global Surface Parameterization*

Surface parameterization has been a very active research area in the past decade [5]. Parameterization can be viewed as a mapping from a surface in 3D to a 2D canonical domain. 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 angu-

lar distortion) [27,8,21,24,16,19], or equiareal (i.e., no area distortion) [23,28,20]. Hyperbolic parametrization for high genus number surfaces is presented in [17]. Spherical parametrization for genus zero surfaces are introduced in [6,9]. In sharp contrast to the above parameterization methods, which build the map between the surface and one of the three canonical domains (sphere, Euclidean disk, or hyperbolic disk), Tarini *et al.* pioneered the concept of polycube maps, which has the same topology of the input mesh and also mimics its rough geometry. Thus, polycube can induce the map which minimizes both the angular distortion and area distortion [29]. Tarini *et al.* demonstrated that polycube maps naturally lead to a seamless texture mapping method that is simple enough to be implemented in currently available graphics hardware [29].

3. Construction of Polycube Maps

In this section, we explain in details our algorithm for constructing affine atlas using polycube maps for surfaces of arbitrary topology. The key difference between the techniques employed in [29] 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.

3.1. *Riemannian Uniformization Metric*

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.

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.

In essence, Riemannian metric determines the length, area, curvature and differential operators on S. When the Riemannian metric is conformally deformed, these geometric quantities will be changed accordingly. Suppose $\mathbf{g}$ is changed to $\bar{\mathbf{g}} = e^{2u}\mathbf{g}$. Then the Gaussian curvature will become

$$\bar{K} = e^{-2u}(-\Delta u + K), \tag{1}$$

where Δ is the Laplacian-Beltrami operator under the original metric $\mathbf{g}$. The geodesic curvature will become

$$\bar{k} = e^{-u}(\partial_{\mathbf{n}} u + k), \tag{2}$$

where $\mathbf{n}$ is the tangent vector orthogonal to the boundary. According to Gauss-Bonnet theorem, the total curvature is

$$\int_S K dA + \int_{\partial S} k ds = \int_{\bar{S}} \bar{K} d\bar{A} + \int_{\partial \bar{S}} \bar{k} d\bar{s} = 2\pi\chi(S), \tag{3}$$

where $\chi(S)$ is the Euler characteristic number of S and ∂S is the boundary of S.

Riemann uniformization theorem [18] states that for any surface S, there exists a unique conformal metric, such that it induces constant Gaussian curvature $\bar{K}$ and zero geodesic curvature $\bar{k}$.

$$\bar{K} = \begin{cases} +1, & \chi(S) > 0 \\ 0, & \chi(S) = 0 \\ -1, & \chi(S) < 0 \end{cases} \tag{4}$$

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

We compute the uniformization metric with heat flow method [9] for genus zero surfaces, holomorphic 1-form method [8,16] for genus one surfaces, and hyperbolic Ricci flow method [17] for surfaces with genus greater than one.

In the followings, 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:

(i) 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.

(ii) 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., $\phi_M : M \to D_M$.

(iii) Compute the uniformization metric of P and embed P in the canonical domain D_P, i.e., $\phi_P : P \to D_P$.

(iv) Construct the map $\phi_{D_M \to D_P} : D_M \to D_P$.

(v) 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 (5).

$$\begin{array}{ccc} M & \xrightarrow{\phi_{M\to P}} & P \\ \downarrow{\scriptstyle \phi_M} & & \downarrow{\scriptstyle \phi_P} \\ D_M & \xrightarrow[\phi_{D_M\to D_P}]{} & D_P \end{array} \tag{5}$$

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. *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. We use the heat flow method to construct conformal maps between a closed genus zero surface and the unit sphere $\mathbb{S}^2$ [9]. The idea is that, for genus zero closed surfaces, conformal maps are equivalent to harmonic maps.

Let $\phi : M \to \mathbb{S}^2$ denote the spherical mapping. The harmonic energy is defined as

$$E(\phi) = \int_M < \nabla\phi, \nabla\phi > dA, \tag{6}$$

where $<,>$ is the inner product in R^3. The critical point of the harmonic energy is the harmonic map. Define the normal component of the Laplacian as

$$(\Delta\phi)^{\perp} = < \Delta\phi, \mathbf{n}\circ\phi > \mathbf{n}, \tag{7}$$

where $\mathbf{n}$ is the normal of $\phi(M)$. If ϕ is the harmonic map, then the tangent component of Laplace-Beltrami operator vanishes, i.e.,

$$\Delta\phi = (\Delta\phi)^{\perp}. \tag{8}$$

Therefore, we can diffuse ϕ to harmonic map by the heat flow method:

$$\frac{d\phi}{dt} = -(\Delta\phi - (\Delta\phi)^{\perp}). \tag{9}$$

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 Isidore Horse 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

$$\tau : (x,y,z) \to (\frac{2x}{1-z}, \frac{2y}{1-z}),\ (x,y,z) \in \mathbb{S}^2. \tag{10}$$

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 ∞:

$$\psi_1 = \frac{(z-z_0)(z_1-z_2)}{(z-z_2)(z_1-z_0)}. \tag{11}$$

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

$$\phi_{D_M\to D_P} = \tau^{-1}\circ\psi_2^{-1}\circ\psi_1\circ\tau. \tag{12}$$

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]

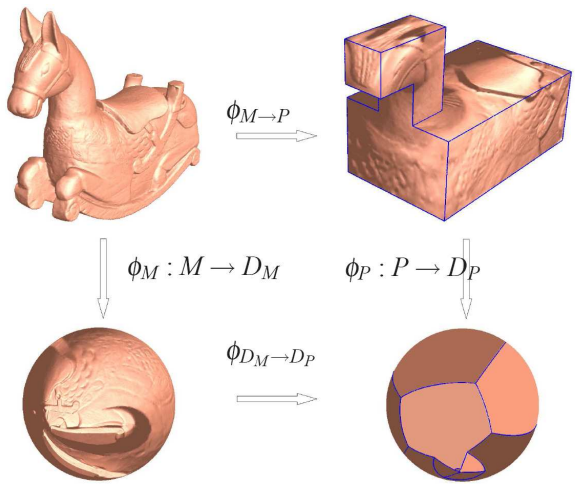


Fig. 2. Conformal mapping of a genus zero surface to the unit sphere induces the genus zero conformal polycube map. Both the original mesh M and the polycube P are conformally mapped to the canonical domains, i.e., $\mathbb{S}^2$, $\mathbb{E}^2$ or $\mathbb{H}^2$. Denote these maps by $\phi_M : M \to D_M$ and $\phi_P : P \to D_P$. By finding the optimal map between D_M and D_P, we get the polycube map $\phi_{M\to P} = \phi_P^{-1}\circ\phi_{D_M\to D_P}\circ\phi_M$.

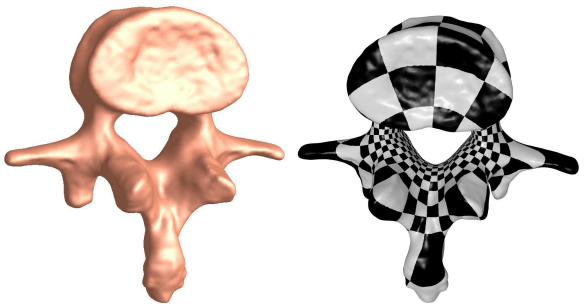


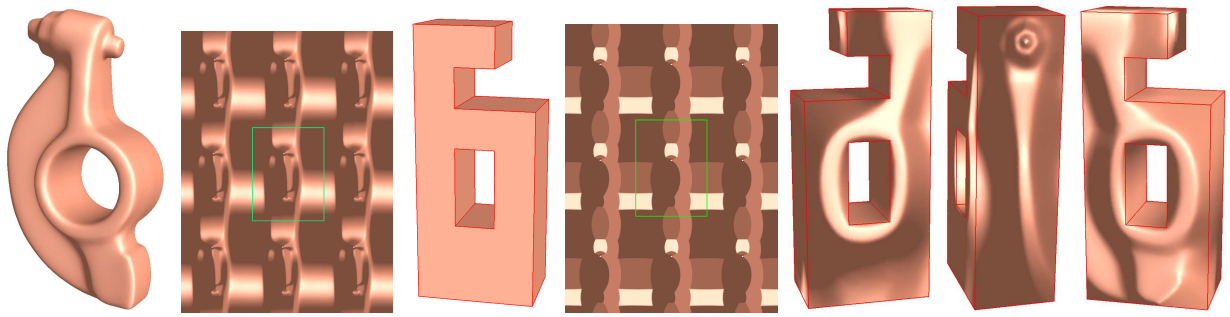
Fig. 3. Holomorphic 1-form ω on genus one surface is well defined everywhere.

3.3. *Genus-one Polycube Map*

Suppose M is a genus one closed surface, ω is a holomorphic 1-form. Then, ω is well-defined everywhere, i.e., there are no zero points as shown in Figure 3.

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.

[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) $\phi : M \to P$

Fig. 4. 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.

Given two arbitrary 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

$$\phi_{M\to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M. \tag{13}$$

Figure 4 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 [8]. 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.

3.4. *High Genus Polycube Map*

Given a high genus surface with simple geometry like the 3-hole torus model shown in Figure 1, the polycube map can be constructed using the techniques in [29]. However, for surfaces with complicated geometries like the model in Figure 5, the direct projection techniques in $\mathbb{R}^3$ hardly generate bijective maps. To avoid these difficulties, we use hyperbolic parameterization method instead.

3.4.1. *Hyperbolic Ricci Flow*

Hyperbolic Ricci flow is introduced in [17]. A circle packing on a mesh associates 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 metric is define as (M,Φ,Γ). A *discrete conformal mapping* $\tau : (M,\Gamma,\Phi) \to (M,\overline{\Gamma},\Phi)$ solely changes the vertex radii Γ, but preserves the intersection angles Φ.

Given the circle packing metric, the length l_{ij} associated with the edge e_{ij} is computed using the hyperbolic cosine law.

$$\cosh l_{ij}^2 = \cosh\gamma_i \cosh\gamma_j + \sinh\gamma_i \sinh\gamma_j \cos\phi_{ij}, \tag{14}$$

where ϕ_{ij} is the intersection angle between two circles associated at v_i and v_j with radius γ_i and γ_j respectively.

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

$$K_i = 2\pi - \sum_{f_{ijk}\in F} \theta_i^{jk}, \qquad v_i \notin \partial M, \tag{15}$$

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

$$K_i = \pi - \sum_{f_{ijk}\in F} \theta_i^{jk}, \qquad v_i \in \partial M. \tag{16}$$

Then the hyperbolic Ricci flow is defined as

$$\frac{\partial \gamma_i}{\partial t} = -\sinh\gamma_i K_i \tag{17}$$

It can be proven that discrete Ricci Flow is convergent to the uniformization metric and the convergence rate is exponential [1][17].

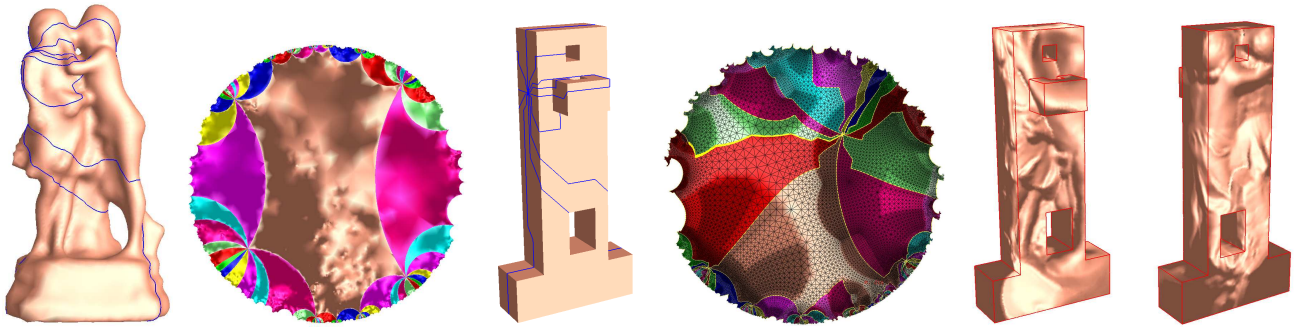
3.4.2. *Hyperbolic Embedding*

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

$$ds^2 = \frac{dzd\bar{z}}{(1-\bar{z}z)^2}. \tag{18}$$

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

$$w = e^{i\theta}\frac{z - z_0}{1 - \bar{z}_0}, \tag{19}$$



(a) (b) (c) (d) (e) (f)

Fig. 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 of the polycube are drawn in blue in (c), (d) shows the isometric embedding of its universal covering space on the Poincaré hyperbolic disk. By establishing the correspondence between the fundamental domains, we construct the polycube map (shown in (e) and (f)) between (a) and (c).

where z_0 is an arbitrary point inside the unit disk.

To embed M into Poincaré disk, we need to compute the canonical homology basis, which is a set of $2g$ curves $\{a_1, b_1, a_2, b_2, \ldots, a_g, b_g\}$ satisfying the following criteria:

(i) All the curves meet at a single base point, v.

(ii) Each pair of curves $\{a_i, b_i\}$ algebraically intersect each other exactly once.

(iii) No curve in another pair $\{a_j, b_j\}$ algebraically intersects either of a_i, b_i.

We slice the mesh M along $\{a_i, b_i\}_{i=1}^{g}$ to form the fundamental domain D whose boundary ∂D is

$$\partial D = a_1 b_1 a_1^{-1} b_1^{-1} \cdots a_g b_g a_g^{-1} b_g^{-1}.$$

Then the canonical homology basis are mapped to geodesics on the Poincaré disk. Figure 6 illustrates the canonical homology basis and the hyperbolic embedding with the uniformization metric for a genus 2 model.

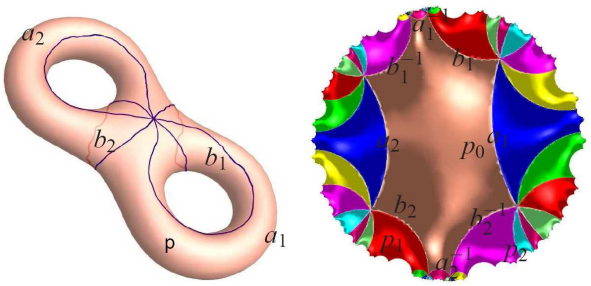


Fig. 6. A genus two surface with a set of canonical fundamental group generators $\{a_1, b_1, a_2, b_2\}$ is shown on the left. A finite portion of its universal covering space is shown on the right. Different fundamental domains are drawn in different colors. The boundary of each fundamental domain 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.

3.4.3. *Constructing the Polycube Map*

In order to find the map between M and P, we compute their hyperbolic parameterizations by solving the discrete hyperbolic Ricci flow in (17). 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

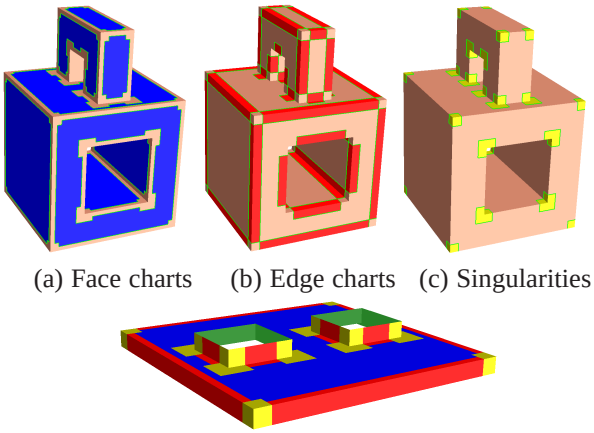
$$\phi_{M \to P} = \phi_P^{-1} \circ \phi_{D_M \to D_P} \circ \phi_M. \tag{20}$$

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

3.5. *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 of 90 degrees. 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 structure. Figure 7 highlights face and edge charts of a polycube. The extraordinary points are colored in yellow.

In [10], 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 view of one face chart and its associated edge charts and singularities

(e) Face chart (f) Transition function

Fig. 7. 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. Hierarchical Surface Reconstruction Using Polycube T-Splines

After constructing the domain manifold and affine atlas of the original model by computing the polycube maps (section 3), we are now ready to generalize T-spline from planar domains to manifold domains via affine structure. This will enable the automatic reverse engineering from polygonal models initially acquired to a more compact spline representation with high accuracy.

4.1. *T-splines Via Polycube Maps*

The key advantage for defining T-spline over polycube maps 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.

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 [25,26]. 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:

$$\mathbf{F}_i(\mathbf{u}) = \sum_j \mathbf{c}_j B_j(\phi_i(\mathbf{u})), \mathbf{u} \in U_i, \tag{21}$$

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:

(i) 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.

(ii) The function value is the partition of unity of the spline patches in the above chart(s), i.e.,

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

4.2. *Least-Square Fitting and Hierarchical Refinement*

We now discuss the problem of finding a good approximation of a given polygonal mesh S with vertices $\{\mathbf{p}_i\}_{i=1}^m$ by a manifold T-spline. We assume that the polygonal mesh S has been normalized to be inside the unit cube centered at the origin. A commonly-used technology is to minimize a linear combination of interpolation and fairness functionals, i.e.,

$$\min E = E_{dist} + \lambda E_{fair}. \tag{22}$$

The first part is

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

where $\mathbf{u}_i \in M$ is the parameter for $\mathbf{p}_i$, $i = 1, \ldots, m$.
The second part E_{fair} in (22) is a smoothing term. A frequently-used example is the thin-plate energy,

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

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

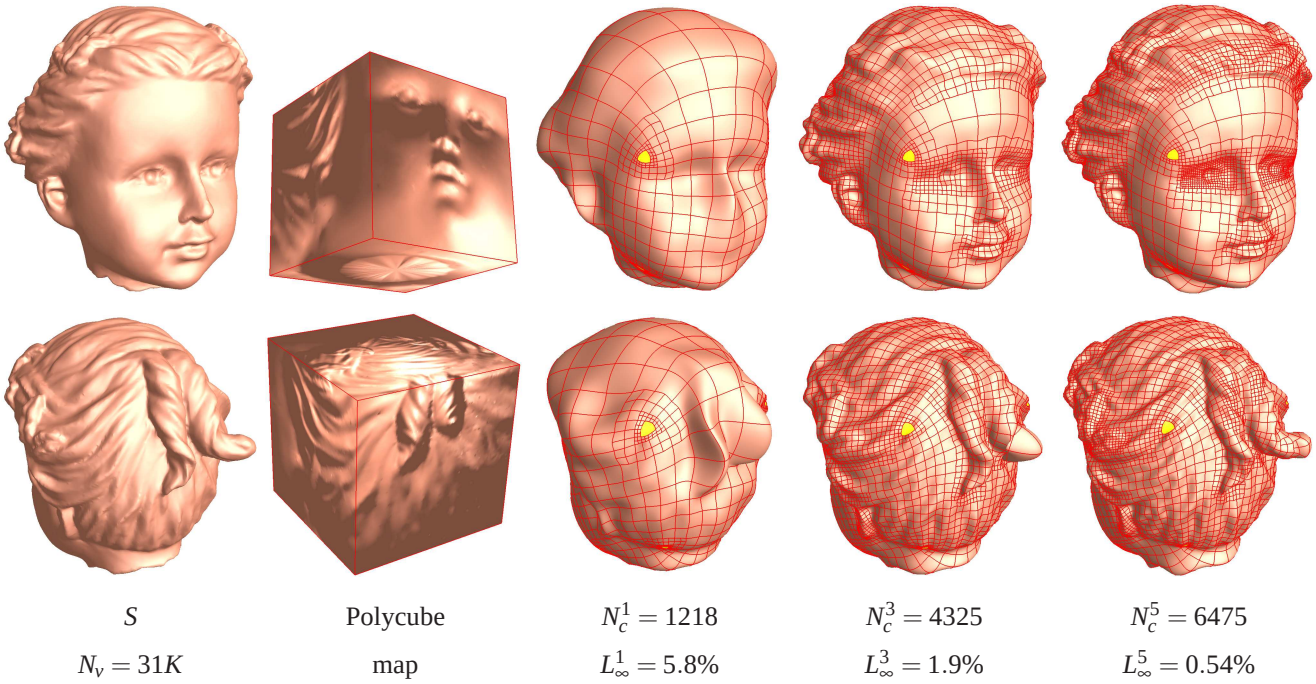


Fig. 8. Hierarchical surface reconstruction of Polycube T-splines. N_c^i and L_∞^i are the number of control points and maximal fitting error in iteration i. N_v is the number vertices in the input polygonal mesh S. The input data is normalized to a unit cube.

We solve Equation (22) for unknown control points using Conjugate Gradient method. The value and gradient of the interpolation functional and fairness functional can be computed straightforwardly.

In our method, we control the quality of the manifold T-spline spline 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 8 shows the whole procedure of hierarchical fitting of T-splines. For example, the initial spline of the Head model(Figure 8) 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 9), 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 [10], Gu *et al.* proved that manifold splines *MUST* have singularities if the domain manifold is closed and not a torus. The number of extraordinary points of the domain manifold via conformal structures and polycube maps are different. Given the surface M with genus g and b boundaries, the number of zero points of the holomorphic 1-form is fixed, i.e., $|2g-2+b|$. Using polycube maps, the number of extraordinary points depends on the geometry of the polycube, i.e., each corner is a singularity.

Although the singularities are just points on the domain manifold, in practice, we have to remove these points and their 1-ring or 2-ring neighbors. As a result, the holes are unavoidably in the spline surface. Thus, we need to find a blending surface patch to fill the holes smoothly. In our implementation, we use a cubic triangular spline to fill each hole such that the surface is C^2 inside and G^1 along the boundaries of the hole. The reason that we choose triangular B-spline [11] is its flexibility in the domain construction and its potential to match with any number of sides of holes.

Thus, our goal is to solve the following optimization problem:

$$E(s) = \iint_\Omega (\frac{\partial^2 s}{\partial u^2})^2 + 2(\frac{\partial^2 s}{\partial u \partial v})^2 + (\frac{\partial^2 s}{\partial v^2})^2 du dv. \qquad (23)$$

where s is the triangular B-spline surface, and Ω is the parametric domain of s. Our strategy to fill the hole is to find s solving the following minimization problem:

$$min\{E(s) : s|_{\partial\Omega} = f,\ \frac{\partial s}{\partial u} \times \frac{\partial s}{\partial v}|_{\partial\Omega} = n\}. \qquad (24)$$

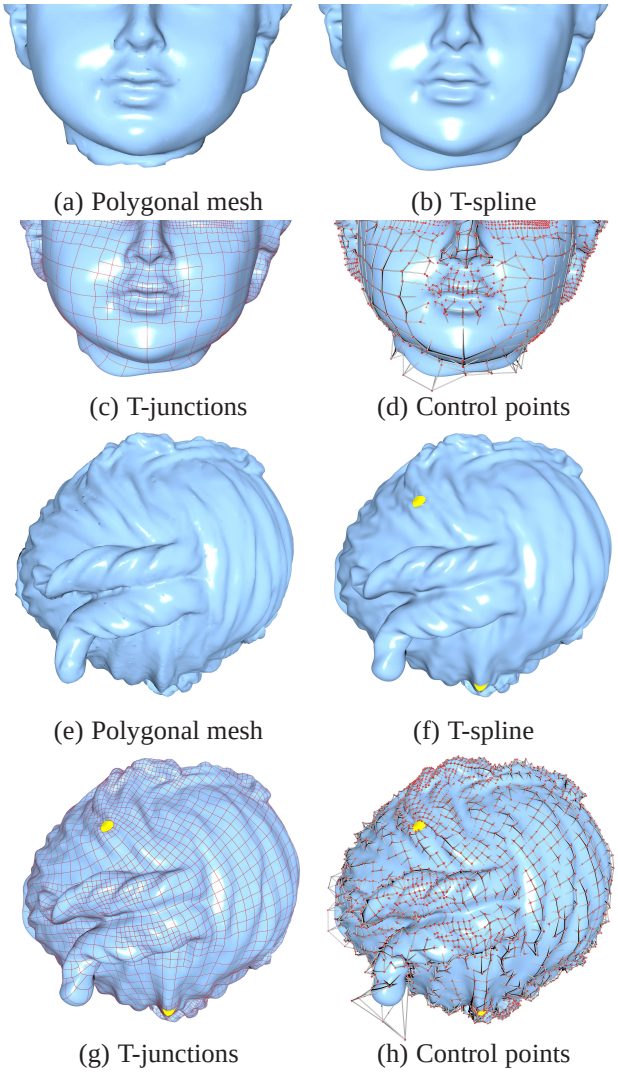


Fig. 9. Close-up views of the reconstructed details. Our hierarchical surface reconstruction algorithm can faithfully reconstruct the details in the original model. (a) and (e) show the original polygonal model. (b) and (f) show the T-spline surfaces of C^2 continuity. (c) and (g) highlight the T-junctions on the spline surfaces. (d) and (h) show the splines overlaid by the control points.

where f and n are the boundary positions and normals.

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

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

Therefore, Equation (24) is a linear least-square problem with linear constraints, which can be solved easily using Lagrange Multiplier method. Figure 10 demonstrates the procedure pipeline to handle the extraordinary points on the Rocker Arm model.

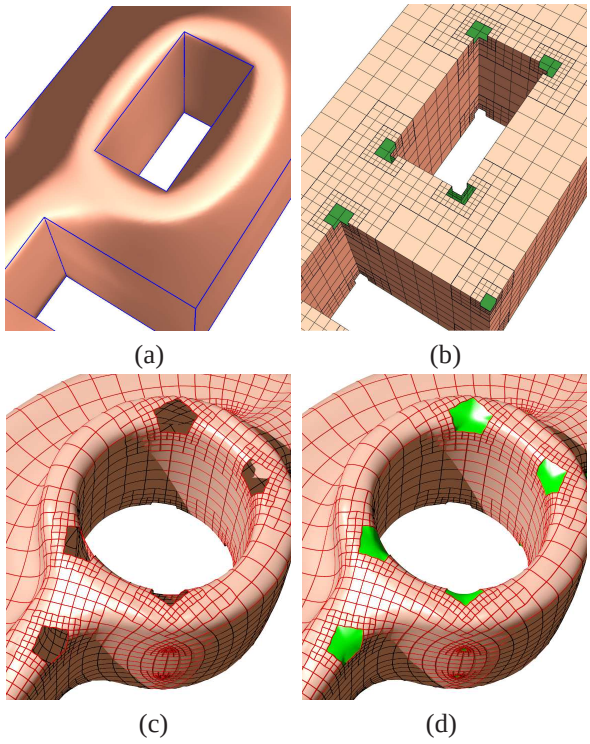


Fig. 10. Handing the extraordinary points of the manifold T-spline whose affine atlas is constructed using polycube maps, where all the corners are extraordinary points (shown in (a)). (b) shows the domain manifold after removing all the corners. (c) shows the open manifold T-spline surface with many holes. For each hole, we construct a cubic triangular *B*-spline surface which minimizes the thin-plate energy functional (24) and satisfies the boundary condition. (d) shows the final result after hole-filling (hole areas are all colored in green).

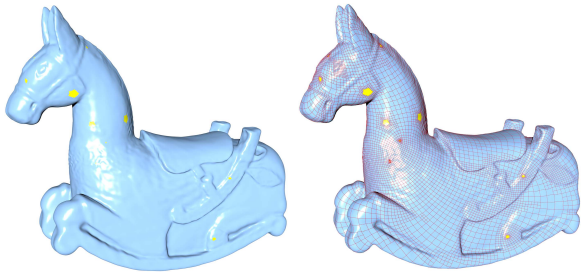
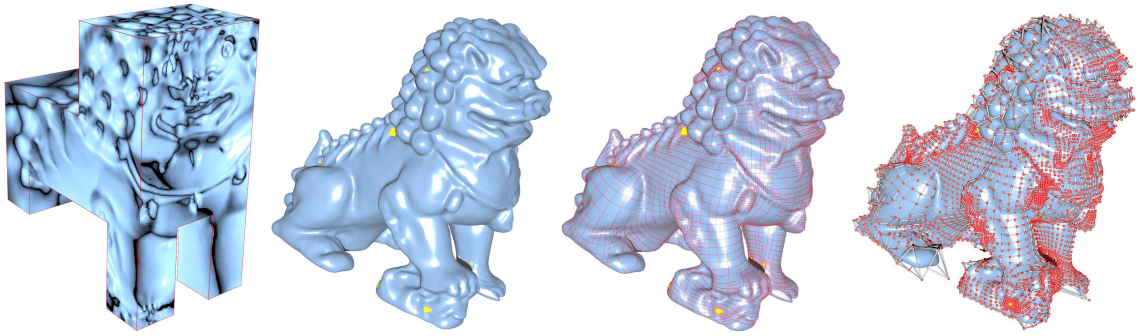


Fig. 11. Polycube T-splines for the Isodore Horse model.

4.4. *Discussions*

This subsection compares the T-splines constructed using conformal structure [15] and polycube map, respectively. From the chart-relation's point of view, these methods differ in three aspects, the number and the locations of singularities, the angle/area distortion, and the type of transition functions. Each method has its own merits and users may choose one or another depending on their specific application needs. Table 1 summaries the salient differences between these methods.

Conformal structure induces the affine structure with the fixed number of extraordinary points, i.e., $|2g-2+b|$. For genus-zero surfaces, we usually intentionally cut two boundaries on



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

Fig. 12. Polycube T-splines for the Chinese Dragon model.

Table 1
Comparison of the methods to compute affine structures. g, genus of the domain manifold M; b, number of boundaries of M.

Method	# of singularities	Location of singularities	Area distortion	Angle distortion	Transition function
Conformal structure	$\|2g-2+b\|$	difficult to control	large on extruding parts	no	translation
Polycube map	many	easy to control	low	low	translation and 90 degree rotation

the model. Note that, we do not modify the geometry of the original model, the number of extraordinary points drop to zero. Although conformal structure preserves the angles very well, they inevitably introduce large area distortion if the model has some long, extruding parts. These large area distortions usually make the spline construction very difficult, since we need to introduce more control points in such areas. The transition functions of the affine atlas via conformal structure is simply the translations, which facilitates the implementation of T-splines on manifolds. The valence of extraordinary points of T-splines via conformal structure is eight, i.e., the hole is sixteen-sided.

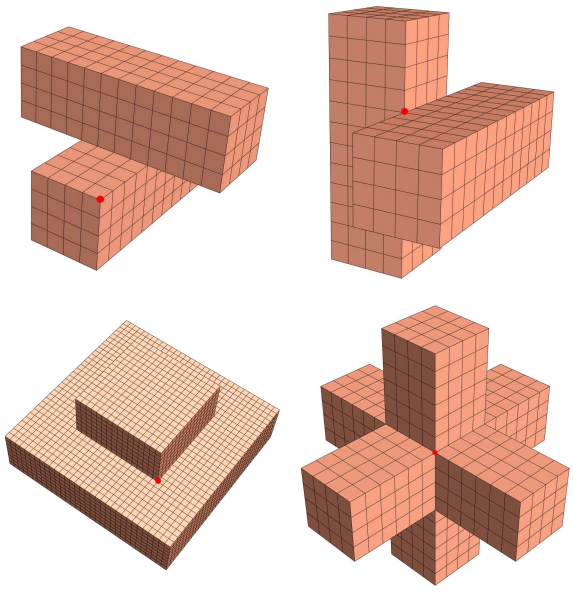


Fig. 13. Extraordinary point (marked in red) with valence 3, 4, 5 and 6.

Polycube maps are ideal to reduce both the area and angle distortion in the affine atlas, as shown in the 3-hole torus models in Figure 1. Thus, it facilitates the spline construction procedures. However, the side-effect to reduce the area distortion is to introduce more extraordinary points simultaneously. Usually, the lesser the area distortion, the more number of extraordinary points. The transition functions of the affine atlas via polycube maps is the composition of translation and 90 degree rotation. The valence of singularities of T-spline via polycube map is three, four, five or six, thus, the hole is six, eight, ten or twelve-sided (see Figure 13).

4.5. *Experimental Results*

Our prototype system is implemented in C++ on Windows platform. We built a complete system for computing the conformal structures, the polycube maps and 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 2. Figure 8 illustrates the hierarchical surface reconstruction. As shown in Figure 8 and 9, we can get high-quality spline surfaces by gradually increasing the number of control points. More complicated models are shown in Figure 11, Figure 12, and Figure 14. The results demonstrate both the theoretic rigor and feasibility in practice for methodologies and computational techniques.

5. Conclusion

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

Table 2
Statistics of test examples. N_s, # of singularities; N_c, # of control points; *rms*, root-mean-square error.

Object	genus	N_s	N_c	*rms*
Head (Figure 8, 9)	0	8	6475	0.05%
Bimba (Figure 14)	0	16	10964	0.07%
Buddha (Figure 14)	0	16	11067	0.04%
Rockerarm (Figure 14)	1	24	4132	0.03%
3-hole Torus (Figure 1)	3	32	5180	0.02%
Isidore Horse (Figure 11)	0	20	12158	0.07%
Chinese Dragon (Figure 12)	0	28	11335	0.07%
Ramesses (Figure 14)	0	24	9874	0.04%

parametric domain. The use of polycubes for spline surface definition and construction is the first attempt to take advantage of the rectangular structure defined by 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 also articulate our strategy for the hole-filling procedure. 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 data fitting for reverse engineering and shape presentation, we foresee a broader application scope in solid computing, shape analysis, data compression, FEM-based dynamic simulation, and virtual prototyping in CAD environments.

Acknowledgements

The models are courtesy of the AIM@SHAPE Shape Repository and Cyberware Inc. Ying He is partially supported by NTU/SUG 19/06 and AcRF RG69/07. Hongyu Wang, Xin Li and Hong Qin are partially supported by NSF IIS-0326388, IIS-0601168, and IIS-0710819. Xianfeng Gu is partially supported by NSF CCF-0448399, DMS-0528363, DMS-0626223 and NSFC-60628202.

References

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

[2] COTRINA, J. AND PLA, N.: 2000. Modeling surfaces from meshes of arbitrary topology. *Computer Aided Geometric Design 17*, 643–671.

[3] COTRINA, J., PLA, N. AND VIGO, M.: 2002. A generic approach to free form surface generation. In *Proceedings of the seventh ACM symposium on Solid modeling and applications*, 35–44.

[4] DENG, J., CHEN, F., AND FENG, Y. 2006. Dimensions of spline spaces over T-meshes *Journal of Computational and Applied Mathematics 194*, 2, 267–283.

[5] 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.

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

[7] GRIMM, C. AND HUGHES, J.F.: 1995. Modeling surfaces of arbitrary topology using manifolds. In *SIGGRAPH*, 359–368.

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

[9] 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.

[10] GU, X., HE, Y., AND QIN, H. 2006. Manifold splines. *Graphical Models 68*, 3, 237–254.

[11] HE, Y. AND QIN, H. 2004. Surface reconstruction with triangular B-splines. In *Proceedings of GMP '04*, 279–290.

[12] HE, Y., GU, X. AND QIN, H. 2005. Rational spherical splines for genus zero shape modeling. In *Proceedings of SMI '05*, 82–91.

[13] 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.

[14] 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.

[15] HE, Y., WANG, K., WANG, H., GU, X., AND QIN, H. 2006. Manifold T-spline In *Proceedings of GMP '06 (LNCS Vol. 4077)*, 409-422.

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

[17] 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.

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

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

[20] KHODAKOVSKY, A., LITKE, N., AND SCHRÖDER, P. 2003. Globally smooth parameterizations with low distortion. *ACM Trans. Graph. 22*, 3, 350–357.

[21] 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.

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

[23] 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.

[24] RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Transactions on Graphics 25*, 4, 1460-1485.

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

[26] 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.

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

[28] 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.

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

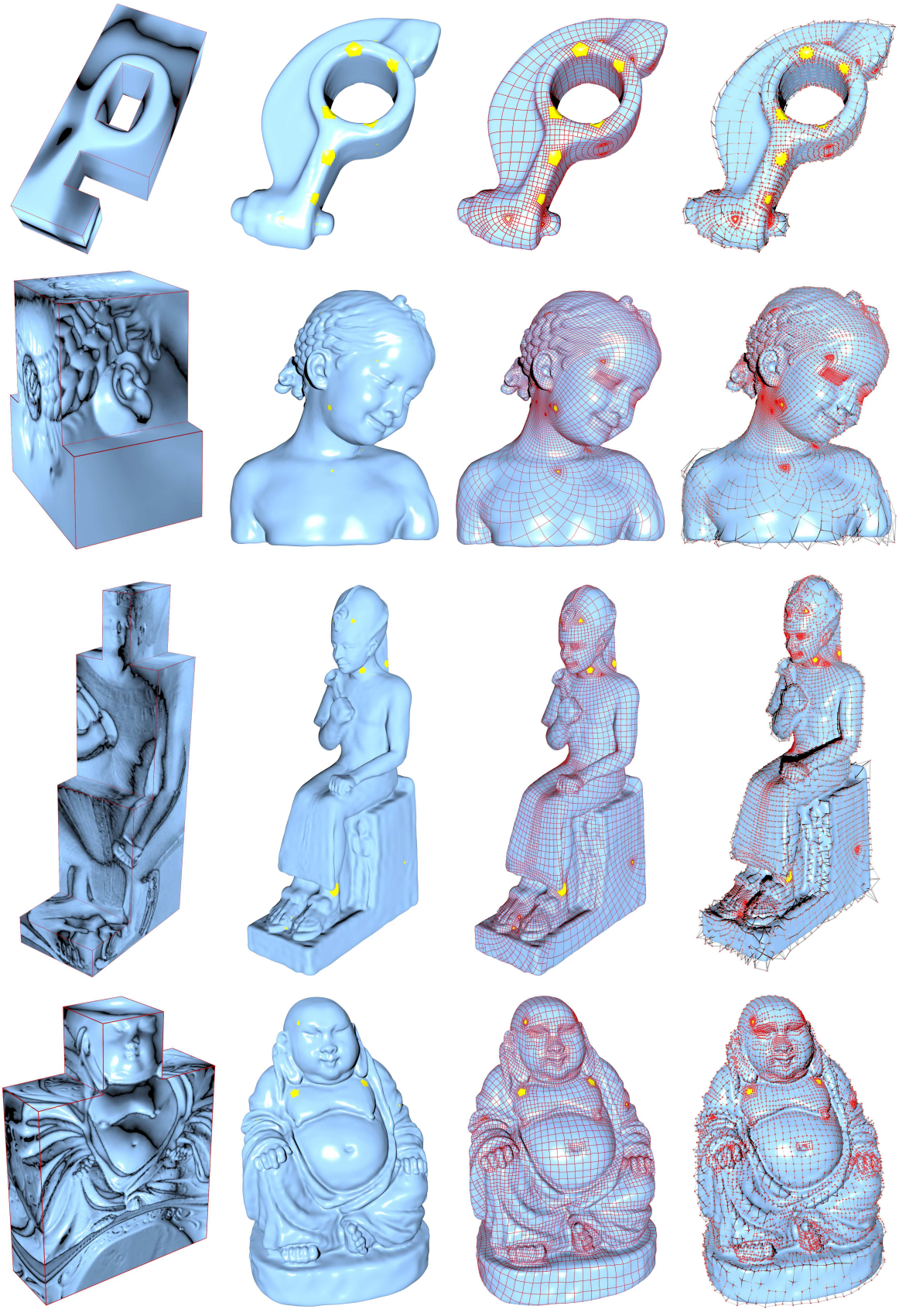
[30] WANG, Y., AND ZHENG, J. 2006. Control point removal algorithm for T-spline surfaces. In *GMP*, 385–396.

[31] YANG, H., FUCHS, M., JÜTTLER, B., AND SCHERZER, O. 2006. Evolution of T-Spline Level Sets with Distance Field Constraints for Geometry Reconstruction and Image Segmentation. In *IEEE Shape Modeling and Applications (SMI)*, 247–252.

[32] YANG, H., AND JÜTTLER, B. 2007. Meshing Non-uniformly Sampled and Incomplete Data Based on Displaced T-spline Level Sets. In *IEEE Shape Modeling and Applications (SMI)*, 251–260.

[33] YING, L., ZORIN, D. 2004. A simple manifold-based construction of surfaces of arbitrary smoothness. *ACM Trans. Graph. 23*, 271–275.

[34] 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) T-spline (c) T-junctions (d) Control points

Fig. 14. Construction of manifold T-splines using polycube maps.