

Stony Brook University



OFFICIAL COPY

The official electronic file of this thesis or dissertation is maintained by the University Libraries on behalf of The Graduate School at Stony Brook University.

© All Rights Reserved by Author.

General Surface Geometric Structures and Their Applications

A Dissertation Presented
by
Miao Jin

to
The Graduate School
in Partial Fulfillment of the
Requirements
for the Degree of
Doctor of Philosophy
in
Computer Science
Stony Brook University

August 2008

Copyright by
Miao Jin
2008

Stony Brook University
The Graduate School

Miao Jin

We, the dissertation committee for the above candidate for
the degree of Doctor of Philosophy, hereby recommend
acceptance of this dissertation.

Xianfeng Gu, Dissertation Advisor
Assistant Professor, Computer Science Department

Hong Qin, Chairperson of Defense
Professor, Computer Science Department

Joseph Mitchell,
Professor, Applied Mathematics and Statistics Department

Jie Gao
Assistant Professor, Computer Science Department

This dissertation is accepted by the Graduate School

Lawrence Martin
Dean of the Graduate School

Abstract of the Dissertation
General Surface Geometric Structures and Their Applications

by
Miao Jin

Doctor of Philosophy
in
Computer Science
Stony Brook University
2008

Geometric structures are natural structures of surfaces, which enable different geometries to be defined on the surfaces coherently and allow general planar algorithmic constructions to be generalized onto the surfaces directly. For example, all oriented surfaces have conformal structure. We can generalize planar texture mapping, texture synthesis, remeshing and mapping algorithms to surfaces based on their conformal structure without angle distortion. Also polar form splines with planar domains can be generalized to manifold splines on the surfaces which admit affine structure and are equipped with affine geometry.

This work presents theoretically rigorous and practically efficient methods for computing general surface geometric structures, including conformal structure, affine structure, hyperbolic structure, real projective structure, and spherical structure. The powerful tool we used is discrete surface Ricci flow. We generalized surface Ricci flow from continuous to discrete setting, and designed a series of algorithms to compute discrete surfaces Ricci flow, which includes discrete Euclidean Ricci flow, discrete hyperbolic Ricci flow, and discrete spherical Ricci flow.

We applied surface geometric structures computed from discrete surface Ricci flow to computer graphics, medical imaging, geometric modeling, and computer vision. We compute globally conformal parametrization for surfaces of general topologies, with less area distortion and control of both the number and location of singularity points; we conformally flatten colon surfaces onto plane, which enhances the navigation of virtual colonoscopy system; we design N-RoSy field on

general surfaces based on flat metric induced from surfaces' conformal structure; we construct manifold spline with single singularity using surface affine structure, which achieves the theoretical minimum of the singularity number; we combine manifold spline and T-spline to polycube T-spline by building polycube map of surface which naturally induces surface affine structure; we compute shape space for general surfaces, where Surfaces are indexed and classified by their conformal structure.

To my parents

Contents

List of Tables	ix
List of Figures	x
Acknowledgements	xix
1 Introduction	1
1.1 Geometry	2
1.2 Geometric Structures on Surfaces	3
1.3 Contribution	5
2 Theoretical Background	7
2.1 Glossary on Differential Geometry	7
2.1.1 Riemannian Metric	7
2.1.2 Conformal Parameterization	8
2.1.3 Gaussian Curvature	8
2.2 Uniformization Theorem	9
2.3 Hyperbolic Space Models	9
2.3.1 Poincaré Model	10
2.3.2 Klein Model	11
2.3.3 Upper Half Plane Model	11
2.4 Fundamental Group and Universal Cover	11
2.5 Surface Ricci Flow	12
3 Theoretical Background on Discrete Surfaces	14
3.1 Discrete Riemannian Metric	15

3.2	Discrete Gaussian Curvature	15
3.3	Discrete Gauss-Bonnet Theorem	16
3.4	Discrete Conformal Deformation	16
3.5	Admissible Curvature Space	17
3.6	Discrete Surface Ricci Flow	18
4	Conformal Structure	20
4.1	Overview	20
4.1.1	Applications for Conformal Structure	22
4.2	Optimal Global Conformal Parametrization Using Discrete One-form	24
4.2.1	Optimal Global Conformal Parameterizations of High Genus Surfaces	25
4.2.2	Optimal Global Conformal Parameterizations of Genus Zero Surfaces	27
4.2.3	Topological Optimization	30
4.2.4	Zero Points Allocation	33
4.3	Surface Global Conformal Parameterization Using Discrete Euclidean Ricci Flow	36
4.3.1	Computing Flat Uniformization Metric	36
4.3.2	Embedding One Domain on Plane	39
4.3.3	Embedding Periodically	40
4.4	Applications	42
4.4.1	Conformal Virtual Colon Flattening	42
4.4.2	Metric-Driven N-RoSy Fields Design	43
5	Affine Structure	48
5.1	Overview	49
5.2	Computing Surfaces Affine Structures	49
5.3	Applications	50
5.3.1	Manifold Spline With Single Singularity	50
5.3.2	User-controllable Polycube T-spline	55
6	Hyperbolic Structure and Real Projective Structure	61
6.1	Overview	61

6.2	Computing Surfaces hyperbolic Structure and Real Projective Structure	62
6.2.1	Computing Hyperbolic Uniformization Metric	63
6.2.2	Computing Fuchsian Group Generators in the Poincaré Disk Model	63
6.2.3	Constructing Hyperbolic Structure	65
6.2.4	Constructing Real Projective Structure	66
6.3	Application: Shape Space	66
6.3.1	Overview	66
6.3.2	Conformal Equivalence	67
6.3.3	Geodesic Spectrum	72
6.3.4	Teichmüller shape space Coordinates	80
7	Spherical Structure and Topological Structure	87
7.1	Spherical Structure	87
7.2	Topological Structure	88
7.3	Application: Topology-based Surface Mapping	88
8	Performance Analysis	92
8.1	Convergence	92
8.2	Speed and Time Complexity	92
8.3	Conformality	95
8.4	Accuracy	95
9	Summary and Future Research	98
9.1	Summary	98
9.2	Future Research	99
9.2.1	Harmonic Volumetric Parameterization Using Green’s Functions on Star Shapes	99
9.2.2	Discrete Curvature Flow for 3-Manifolds	100
	Bibliography	103

List of Tables

1	Geometric Structures on Surfaces	5
2	Geodesic Spectrum of Genus One Surfaces.	76
3	Geodesic Spectrum of Genus Three Surfaces.	77
4	Geodesic Spectrum of Genus Two Surfaces.	78
5	Comparison of Coordinates of Vase Model with Different Densities. The dimension of Teichmüller space coordinates for closed genus two surfaces is seven.	84
6	Distances between genus two surfaces in Techmüller space.	85
7	The sorted distances between teapot7 and other genus two models in Techmüller space. Here we only show the closest ones.	86
8	The sorted distances between eight and other genus two models in Techmüller space. Here we only show the closest ones.	86
9	Time of computing discrete Ricci flow.	94
10	Accuracy Testing for Hyperbolic Layout in Fig. 55. Möbius transformation M_0 moves domain in (d) to (c); M_1 moves (e) to (c); M_2, M_3 , and M_4 are deck transformations computed from different edges on the same boundary segment.	97

List of Figures

1	Manifold: The manifold is covered by a set of charts (U_α, ϕ_α) , where $\phi_\alpha : U_\alpha \rightarrow \mathbb{R}^2$. If two charts (U_α, ϕ_α) and (U_β, ϕ_β) overlap, the transition function $\phi_{\alpha\beta} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is defined as $\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha^{-1}$	4
2	Cocycle condition for transition functions.	4
3	Conformal parameterizations map infinitesimal circles to infinitesimal circles and preserve the intersection angles among the circles. Here we use finite circles to approximate. Tangency of circles are preserved.	8
4	Uniformization Theorem: all surfaces with Riemannian metric can be conformally embedded onto three canonical spaces: sphere, plane and hyperbolic space.	10
5	Circle Packing Metric (a) Circle packing metric on one triangle. (b) Circle packing metric on planar triangulations.	16
6	Max Planck Head Model. (a).Most uniform conformal parametrization, energy: 1.605e-5 (b). Least uniform conformal parametrization, energy: 4.231e-5	28
7	David Head Model. Möbius transformation from the unit disk to itself.	30
8	Cactus Model. (a).Most uniform parametrization with cactus model (b). Most uniform parametrization after one slice on the top of the left branch (c). Most uniform parametrization after another slice on the top of the right branch.	31

9	Bunny Model With Topological Optimization. (a). Conformal Factor is color encoded into bunny (b). Topological Optimization: three slices in bunny (c). Most Uniform Global Conformal Parametrization, energy: $0.37e-4$ (d). Least Uniform Global Conformal Parametrization, energy: $2.1e-4$	31
10	Genus two eight model with zero points locating at different positions.	33
11	Genus two eight model with zero points locating at different positions.	33
12	David Half Body Model With Zero Points Hidden. (a). One zero point hidden under left armpit (b). The other zero point hidden under right armpit	34
13	Human Body Model.	34
14	Camel Model and Lion Model.	35
15	Horse Model and Hull Model.	35
16	(a) Horse model with four boundaries on its hoofs. (b) All curvatures are pushing to its four boundaries, and all interior vertices are 'flat', with zero Gaussian curvature. (c) The horse is embedded on the plane with the new metric. (d) All curvatures are concentrated on one singularity vertex, which is marked with red. (e) The horse is embedded on the plane with the new metric.	37
17	(a) Genus one kitten model marked with a set of canonical fundamental group generators which cut surface into a topological disk with four sides: a, b, a^{-1}, b^{-1} . (b) One period is conformally Embedded onto plane, marked with four sides. (c) One translation moves the side b of one copy of the period matching the side b^{-1} of itself. (d) Another translation moves the side a of another copy of the period matching the side a^{-1} of itself. (e) A portion of tessellate of the kitten in planar domain. (f) The Euclidean coordinate of each vertex in (b) is used as texture coordinate for the kitten model in (a), with check board texture.	40

18	Conformal Virtual Colon Flattening: (a) The colon wall (b) The holomorphic one-form on the colon surface is illustrated using texture-mapping with a checker board image.	43
19	A flattened image for a whole colon data set is shown in three images, with polyps marked with small circles.	44
20	Algorithm pipeline (a). User specifies the desired singularities with both positions and indices. Here one singularity at the blue point with the index -2 . The curves are homotopy group basis. (b) We compute a flat metric, the curvature at the singularity is -4π , everywhere else 0. The surface is cut along the base curves and flatten to the plane. Note that the boundaries of the same color can match each other by a rigid motion. (c) Parallel vector field. The field has discontinuities along the red curve. (d) Compute a harmonic 1-form to compensate the holonomy. (e) The smooth vector field after rotation compensation. (f)(g) User inputs geometric constraints (red arrows) to guide the direction of the field, then the field is modified from (f) to (g).	45
21	46
22	Celtic knots designed surfaces	46
23	Pen-and-ink sketching of bimba before ((a) and (b)) and after editing ((c) and (d)). The hatch directions follow the natural directions better (e.g. neck,arm).	47

29	(a) and (b): Corner points are marked on Buddha model, red ones with $\pi/2$ target Gaussian curvature, and green ones with $-\pi/2$ target Gaussian curvatures. (c) and (d): Geodesics between corner points are marked with sharp edges, which are computed using Dijkstra's algorithm with computed conformal metric as edge lengths.	58
30	(a) and (b): One patch from Buddha model after partition, which corresponds to one face of the polycube. The one on the left is the quadrilateral before rectification, the one on the right is the rectangle after rectification. (c) and (d): User-controllable Polycube Map for Buddha Model.	59
31	(a) User-controllable Polycube map (b) Polycube T-spline (c) T-junctions on polycube spline (d) Close-up of control points	59
32	(a) User-controllable Polycube map (b) Polycube T-spline (c) T-junctions on polycube spline (d) Close-up of control points	60
33	(a) Genus two vase model marked with a set of canonical fundamental group generators which cut surface into a topological disk with eight sides: $a_1, b_1, a_1^{-1}, b_1^{-1}, a_2, b_2, a_2^{-1}, b_2^{-1}$. (b) One period is conformally Embedded onto Poincaré disk with uniformization metric, marked with eight sides. (c) One Möbius transformation moves the side b_1 of one copy of the period matching the side b_1^{-1} of itself. (d) Eight Möbius transformations move sides of eight copies of the period matching the corresponding sides of itself. (e) A portion of tessellate of the vase in Poincaré disk. (f) Zoom in of the tessellation, showing no folding or overlapping. (g) Considering an Euclidean unit disk which is coincide with the Poincaré disk, the Euclidean coordinate of each vertex in (b) is used as texture coordinate for the vase model in (a), with check board texture.	62
34	(a) Three human faces sharing the same topology (two holes annulus) are conformally mapped to hyperbolic space. (b) Map between genus one kitten model and torus.	69

35	(a) Two groups of deformed teapot models. (b) Corresponding to two paths in Teichmüller space with each point representing a deformed model.	71
36	Conformal structure of the genus zero surface with 3 boundaries are determined by the lengths of the boundaries under the uniformization metric.	74
37	Genus One Models: Kitten model; Teapot model; Rocker Arm model; Torus model; Elk model; Knotty torus Model.	76
38	Genus Three Models: David model; Genus3 model; Three-Hole model; Holes3 model.	77
39	Genus Two Models: (a) Amphora model (b) World Cup model (c) Vase1 model (d) Vase2 model (e) Cup1 model (f) Cup2 model (g) Eight1 model (h) Eight2 model (i) Knotty model (j) Ding model . . .	77
40	(a) Comparison of geodesic spectrum of genus one models. (b) Comparison of geodesic spectrum of genus three models.	79
41	Comparison of geodesic spectrum of genus two models.	79
42	Basic Building Blocks. (a) Building block I: the geodesic lengths of red labeled curves determine its metric. (b) Building block II: the geodesic lengths of red labeled curves determine its metric. (c) Building block III: the geodesic lengths of red labeled curves determine its metric.	80
43	Using building blocks I, II and III to build all surfaces: using building block I and II to build genus one surface with two boundaries; continuously adding building block III to build genus two surfaces with one boundary; continuously adding building block II to build genus two surfaces with two boundaries. Repeating to get all surfaces.	80
44	The geodesic lengths of the set of color labeled curves determine the metric of a genus two surface.	80
45	Same model with different triangulation density: 5k, 10k, 20k and 40k. Comparison of Teichmüller space coordinates with different densities is listed in table 5.	84

46	The dimension of Teichmüller space coordinates for closed genus two surfaces is seven. Here we visualize the Teichmüller space coordinates for teapots listed in table 7.	84
47	David head model mapped to unit sphere using computed spherical uniformization metric.	88
48	Visualization of two different homotopy classes of mappings between the vase and the two-hole torus by color coding. Each mapping is produced by a different homology basis on each surface. (a) and (b) give the mapping between the two-hole torus and the vase with handles right to right, left to left. While we obtain a different mapping between them in (c) and (d) with the handles right to left, left to right.	89
49	Procedure for computing a canonical fundamental domain, M^* . Step 1: Compute an irregular cut graph that will open the surface to a single disk. Step 2: Slice the surface open to a non-canonical fundamental domain, whose 18 edges are segments of homology basis curves. Step 3: Glue copies of the domain to itself along corresponding segments to form the UCS \tilde{M} . Choose a vertex $v \in M$ and a preimage of v , $\tilde{v}_0 \in \tilde{N}$. Trace paths between \tilde{v}_0 other preimages $\tilde{v}_k \in \tilde{M}$ of v , each of which corresponds to a single homology class of curves on M . Step 4: Find a combination of such curves that forms a canonical homology basis. Step 5: Slice the surface along this cut graph, constructing a canonical fundamental domain M^* . The i th handle is sliced open along the curve sequence $a_i, b_i, a_i^{-1}, b_i^{-1}$	90
50	An overview of the surface mapping process. The vase and the two-hole torus are cut open along a common set of curves and parameterized over the canonical fundamental domain M^* . The final mapping is extracted from the overlaid meshes. τ can be used to change the homotopy type of the mapping.	91

51	Surface mapping between horse and lizard. The color-coding shows the mapping of each region, guided by eight user-specified feature curves. Our topology-driven method provides mappings of different homotopy type between the two surfaces as shown in (c) and (d). We show feature curves in red.	91
52	(a) Performance of discrete Euclidean ricci flow: blue curves are with Newton’s method; green curves are with steepest descent method. (b) Performance of discrete hyperbolic ricci flow: blue curves are with Newton’s method; green curves are with steepest descent method.	93
53	Performance of discrete spherical ricci flow: blue curves are with Newton’s method; green curves are with steepest descent method.	93
54	Conformality Testing Discrete Euclidean Ricci flow conformally maps an irregular shape in (a) to a disk in (b). The parameterization is illustrated as the check board texture mapping in (c), where all the corners of the checkers are well-preserved visually. The histogram of the angle ratio, which is defined as the ratio of the original angle value and the new one after mapping, is calculated and shown in (d). The distribution highly concentrates at 1.	95
55	Accuracy Testing for Hyperbolic Layout Frame (a)-(b): a set of fundamental group generators are marked with red on a genus two eight model with $4k$ faces, and three randomly chosen seed faces are marked with red, green, and blue respectively. Frame (c)-(e) show the flattened results on the Poincaré disk using the hyperbolic uniformization metric, with different seed faces. Any two of them only differ a Möbius transformation. Frame (f): randomly chosen Edges marked with different colors can be used to compute the same deck transformation shown in (g).	97

56	Algorithm pipeline. Given the genus zero sheep model, let ∂M be the boundary surface (a). We first construct a conformal polycube map $\phi_0 : \partial M \rightarrow \partial D$ (b). Then we decompose both the 3D model and polycube into three components $M_i, D_i, i = 1,2,3$ (see (c) and (d)). Each of M_i or D_i is a star shape, for example, the sheep head in (e) and (f). Next, we conformally map ∂M_i and ∂D_i to a sphere (see (g) and (h)). Then we compute Green's functions on the spheres. The cut views of the Green's function are shown in (i) and (j). The Green's function induces a one-to-one map between M_i and D_i as shown in (k) and (l). By gluing the segmented components together and improving the continuity along the cutting boundaries, we get the global volumetric parameterization between M and D. (m),(o) and (p) show the hexahedral remeshing results using the constructed volumetric parameterization.	101
57	The boundary surface of a 3-manifold, Thurston's knotted Y. The volume can be treated as a solid ball with three entangled tunnels.	102
58	Embedding the 3-manifold periodically in the hyperbolic space H^3 with different viewpoints.	102

Acknowledgements

I want to express my deep gratitude to my advisor, Professor Xianfeng Gu, for his years of support and encouragement. He led me into the worlds of both graphics and mathematics, and taught me how to set higher goals and to accomplish them.

I would also like to thank Professors Feng Luo, Hong Qin, Arie Kaufman, Joseph Mitchell, Dimitris Samaras, Klaus Mueller, Michael Ashikhmin, Jie Gao, and Qin Lv for valuable suggestions over the years.

I want to thank all the current and past Visualization Lab members for their help. The life at Stony Brook is a really great experience for me.

My work is supported by NSF Career award 0448339, NSF DMS 0528363, and NSF DMS 0626223.

Chapter 1

Introduction

One of the **fundamental problems** in graphics, geometric modeling, computational geometry and visualization is

How to generalize the algorithms designed for planar domains, spherical domains or some other canonical domains systematically and straightforwardly to the surface domains.

For example, in computer graphics, texture mapping and texture synthesis generalize trivial texture mapping or synthesis on planar domains to surfaces with arbitrary topologies to greatly enhance visual effects, and are commonly used in computer games.

Another example is coming from digital geometry processing, transforming processing tasks on planar domains like re-meshing, to the surfaces. Re-meshing provides highly regular tessellations on the mesh, which is greatly preferred for numerical computations.

In computer-aided geometric modeling, traditional splines are defined on planar domains. Manifold splines generalize spline construction from planar domains to manifold domains with general topologies.

In the medical imaging field, conformal brain mapping has been widely used, which generalizes comparison between two unit sphere domains to two human brains by conformally mapping brain surfaces to spheres to facilitate brain registration, fusion, and comparison.

Geometric structures offer theoretic rigorous and practical efficient solutions to this central problem. Discrete surface Ricci flow is a powerful and flexible tool to compute general surface geometric structures.

1.1 Geometry

Different geometries can be defined on the plane \mathbb{R}^2 and each of them studies different invariants under the corresponding transformation group of \mathbb{R}^2 . The most common geometries on the plane are

1. **Euclidean geometry** The transformation group is the rigid motion group and each rigid motion has the form $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$

$$\phi(\mathbf{p}) = O\mathbf{p} + \mathbf{q}, O \in SO(\mathbb{R}, 2), \mathbf{q} \in \mathbb{R}^2, \quad (1)$$

where O is a rotation matrix with determinant $+1$, and $SO(\mathbb{R}, 2)$ represents the 2 dimensional special real rotation matrix group. The invariants are the *distance* between two arbitrary points, angles of corners, collinearity (i.e., all points lying on a line initially still lie on a line after transformation) etc.

2. **Affine geometry**, The transformation group is the affine transformation group,

$$\phi(\mathbf{p}) = A\mathbf{p} + \mathbf{q}, A \in GL(\mathbb{R}, 2), \mathbf{q} \in \mathbb{R}^2, \quad (2)$$

where A is a real matrix with a positive determinant, and $GL(\mathbb{R}, 2)$ denotes the 2 dimensional real general matrix group. The invariants are the collinearity, ratio between distances, and parallelism.

3. **Projective Geometry** The transformation group is the real projective transformation, $\phi \in PGL(\mathbb{R}, 2)$, where $PGL(\mathbb{R}, 2)$ represents 2 dimensional real projective matrix group,

$$\phi(x, y) = \left(\frac{\alpha x + \beta y + \gamma}{\delta x + \epsilon y + \zeta}, \frac{\eta x + \theta y + \kappa}{\delta x + \epsilon y + \zeta} \right), \begin{vmatrix} \alpha & \beta & \gamma \\ \eta & \theta & \kappa \\ \delta & \epsilon & \zeta \end{vmatrix} \neq 0. \quad (3)$$

The invariants are the colinearity and the cross ratio among four points on the line.

Different algorithms in computer graphics, computational geometry, solid modeling, and visualization are based on different geometries. The following are some examples:

1. *Voronoi Diagram* Given a set of points $\{\mathbf{p}_k\} \subset \mathbb{R}^2$, the whole plane is partitioned to cells $\{C_k\}$. A point \mathbf{p} belongs to C_k , where $k = \min_j |\mathbf{p} - \mathbf{p}_j|$. Therefore, the Voronoi diagram is based on planar Euclidean geometry, where distance plays vital role.
2. *Point location* Given a triangulation of the plane and an arbitrary point \mathbf{p} , the point location algorithm will find the unique triangle which contains \mathbf{p} . Suppose \mathbf{p} is contained in a triangle $\Delta\mathbf{p}_0\mathbf{p}_1\mathbf{p}_2$, then every element of the barycentric coordinates of \mathbf{p} (α, β, γ) must be positive, where

$$\mathbf{p} = \alpha\mathbf{p}_0 + \beta\mathbf{p}_1 + \gamma\mathbf{p}_2.$$

It is obvious that barycentric coordinates are affine invariants. Therefore, the point location algorithm solely depends on the affine geometry of the plane.

3. *Line segments intersection* The sweep line algorithm computes all the intersections among a set of line segments on the plane. colinearity and intersection relations are invariant under projective transformation. Therefore, line segment intersection algorithms are based on the projective geometry of the plane.

1.2 Geometric Structures on Surfaces

Surfaces are manifolds. In general, there are no global coordinates. Instead, a surface M is covered by a set of open sets $\{U_\alpha\}$ as shown in Figure 1. Each U_α can be parameterized by a local coordinate system, and a map $\phi_\alpha : U_\alpha \rightarrow \mathbb{R}^2$ maps U_α to its parameter domain. (U_α, ϕ_α) is a local chart for the surface M . A particular point p may be covered by two local coordinates systems (U_α, ϕ_α) and (U_β, ϕ_β) . The transformation of the local coordinates of p in (U_α, ϕ_α) to those in (U_β, ϕ_β) is formulated as the chart transition map $\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha^{-1}$. Transition maps satisfy the cocycle condition (see Figure 2):

$$\phi_{\alpha\beta} \circ \phi_{\beta\gamma} = \phi_{\alpha\gamma}, \forall p \in U_{\alpha\beta} \cap U_{\beta\gamma}$$

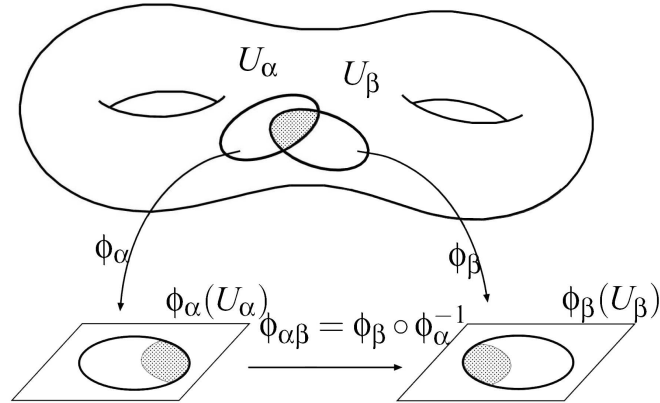


Figure 1: Manifold: The manifold is covered by a set of charts (U_α, ϕ_α) , where $\phi_\alpha : U_\alpha \rightarrow \mathbb{R}^2$. If two charts (U_α, ϕ_α) and (U_β, ϕ_β) overlap, the transition function $\phi_{\alpha\beta} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is defined as $\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha^{-1}$.

All the charts form the atlas $\{(U_\alpha, \phi_\alpha)\}$.

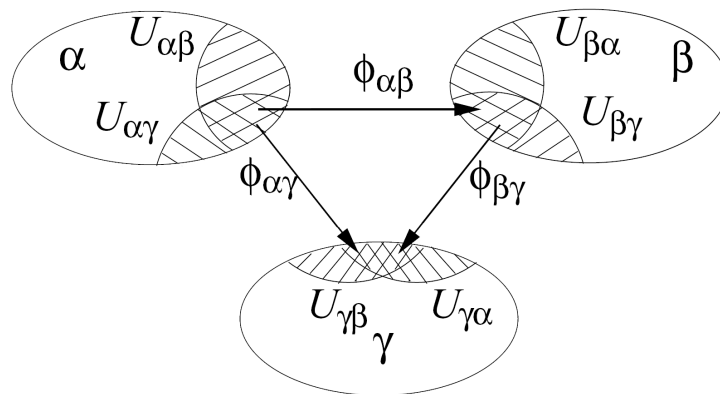


Figure 2: Cocycle condition for transition functions.

If all chart transition maps are rigid motions on \mathbb{R}^2 , then we can discuss the concepts of angle, distance, and parallelism on the surface locally. These geometric measurements can be calculated on one chart, and the results are independent of the choice of the charts. Namely, we can define Euclidean geometry on the surface. Similarly, if all transition maps are affine, then we can define parallelism on the surface. If all transition maps belong to a particular transformation group of \mathbb{R}^2 , we can define the corresponding geometry on the surface. Thurston [48] gave the

(X, G) Structure	Parameter domain X	Trans. group, G	Oriented Metric Surfaces
Topology	\mathbb{R}^2 Plane	Homeomorphisms	Surfaces of arbitrary topology
Differential	\mathbb{R}^2 Plane	Diffeomorphisms	Surfaces of arbitrary topology
Spherical	\mathbb{S}^2 Sphere	Rotation $SO(3)$	Genus zero closed, open
Euclidean	\mathbb{R}^2 Plane	Rigid motion	Genus one closed, open
Hyperbolic	\mathbb{H}^2 Hyperbolic Space	Möbius	High genus closed, open
Affine	\mathbb{R}^2 Plane	Affine $GL(\mathbb{R}, 2)$	Genus one closed, open
Projective	\mathbb{RP}^2 Projective space	Projective	all oriented surfaces

Table 1: Geometric Structures on Surfaces

concept of (X, G) structure, where X is a topological space, and G is a subgroup of the transformation group of X , if M has an atlas $\{(U_\alpha, \phi_\alpha)\}$, such that the parameter domain $\phi_\alpha(U_\alpha) \subset X$ is in space X , and the transition maps $\phi_{\alpha\beta} \in G$ are in G .

Surfaces have rich (X, G) geometric structures, while the existence of a specific geometric structure on a given surface is determined by the surface topology. For example, surfaces with positive Euler numbers have spherical structure; surfaces with zero Euler numbers have affine and Euclidean structures; surfaces with negative Euler numbers have hyperbolic structure. Table 1 illustrates the common geometric structures.

It is a very challenging problem to design rigorous and practical methodology to compute general geometric structures on surfaces. Ricci flow is developed recently in geometric analysis field for the purpose of proving Poincaré conjecture. It offers a powerful tool to conquer this problem.

1.3 Contribution

For theory part, this work brought the concepts of surface geometric structures and surface Ricci flow to engineering fields.

For computation part, this work presented efficient algorithms to compute discrete surface Ricci flow, and a series of algorithms to compute common geometric structures on general surfaces using discrete surface Ricci flow, including conformal structure, affine structure, hyperbolic structure, real projective structure, and spherical structure.

For application part, this work applied general surface geometric structures to computer graphics, computer vision, geometric modeling, and medical imaging, which include:

1. Globally conformal parametrization for surfaces of general topologies based on surface conformal structure, with less area distortion and control of both the number and location of singularity points;
2. Manifold spline construction for closed high genus surfaces with single singularity based on surface affine structure, which achieves the theoretical minimum of the singularity number;
3. Polycube T-spline construction with automatic polycube map based on surface affine structure;
4. Shape space where surfaces are indexed and classified by their conformal structure based on surface hyperbolic structure and Teichmüller theory;

Chapter 2

Theoretical Background

In this section we briefly introduce the major concepts from algebraic topology, differential geometry, Riemann surface to explain geometric structures on surfaces. We limit ourselves to those concepts that are directly relevant to our work. For detailed explanations, we refer readers to [49].

2.1 Glossary on Differential Geometry

2.1.1 Riemannian Metric

Suppose S is a C^2 smooth surface embedded in \mathbb{R}^3 with parameter (u_1, u_2) . A position vector on S is $\mathbf{r}(u_1, u_2)$ and the tangent vector at the point is defined as $d\mathbf{r} = \mathbf{r}_1 du_1 + \mathbf{r}_2 du_2$, where $\mathbf{r}_1, \mathbf{r}_2$ are the partial derivatives of \mathbf{r} with respect to u_1 and u_2 respectively. We call the length of the tangent vector as the *Riemannian metric* and it is represented by the *first fundamental form*, as follows.

$$\begin{pmatrix} du_1 & du_2 \end{pmatrix} \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} \begin{pmatrix} du_1 \\ du_2 \end{pmatrix}, \quad (4)$$

where $g_{ij} = \langle \mathbf{r}_i, \mathbf{r}_j \rangle$. The symmetric matrix $(g_{ij})_{2 \times 2}$ is called the *Riemannian metric* matrix.

2.1.2 Conformal Parameterization

A special parameterization can be chosen to simplify the Riemannian metric. Let $g_{11} = g_{22} = e^{2\lambda}$ and $g_{12} = 0$, such parameter is called the *isothermal coordinates*. In this case, the parameterization \mathbf{r} is conformal, and the analytic behavior on the surface S becomes very simple. For a given surface point, the magnitudes of all directional derivatives of the function are same and they only depend on the parametric position of the point (see Fig 3(b)). Moreover, any intersection angles between two cross lines in the parametric domain is preserved on the surface (see Fig 3(c)). We call λ the conformal factor of the point with the given parameterization.

Suppose $u : S \rightarrow \mathbb{R}$ is a function defined on the surface S , then $e^{2u}\mathbf{g}$ is another Riemannian metric on S . Given arbitrary two tangent vectors at one point, the angle between them can be measured by either \mathbf{g} or $e^{2u}\mathbf{g}$ with the same measurement. Therefore we say $e^{2u}\mathbf{g}$ is *conformal* (or angle preserved) to \mathbf{g} .

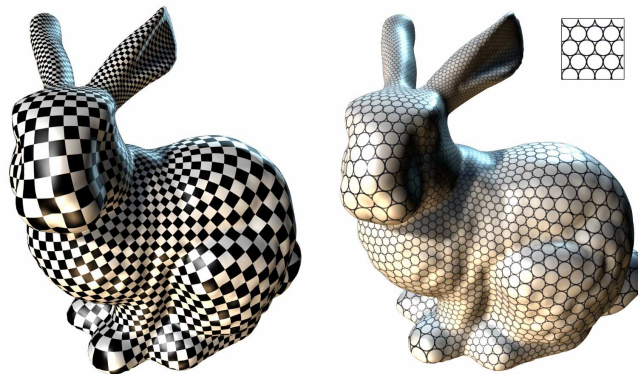


Figure 3: Conformal parameterizations map infinitesimal circles to infinitesimal circles and preserve the intersection angles among the circles. Here we use finite circles to approximate. Tangency of circles are preserved.

2.1.3 Gaussian Curvature

The Gaussian curvature of a point p on a surface S is defined as the ratio between the infinitesimal area around p on S and the infinitesimal area on the unit sphere which is defined by the bunch of normals in the infinitesimal area around

p . The sign of the Gaussian curvature depends on the shape of infinitesimal neighbor of p on S : zero for plane region, positive for elliptic region and negative for hyperbolic region.

Gauss-Bonnet Theorem Gauss-Bonnet theorem explains the connection between the total Gaussian curvatures on a surface S and the topology of S . It tells that how many times the normal field of S covers the unit sphere is strictly related with the topology of S .

$$\int_S K dA + \int_{\partial S} k_g ds = 2\pi\chi(S), \quad (5)$$

while K is the Gaussian curvature on S , k_g the geodesic curvature along boundaries of S , and $\chi(S)$ Euler number of S .

2.2 Uniformization Theorem

Theorem 1 (Uniformization Theorem) *Let (S, \mathbf{g}) be a compact 2-dimensional Riemannian manifold with Euclidean metric \mathbf{g} , then there is a unique metric $\bar{\mathbf{g}}$ conformal to \mathbf{g} with constant Gauss curvatures.*

Such a metric is called the *uniformization metric*. According to Gauss-Bonnet theorem 5, the sign of the constant Gauss curvature is determined by the Euler number of the surface. Therefore, all closed surfaces exist unique metrics conformal with original one, and can be isometrically embedded onto three canonical surfaces, the sphere for genus zero surfaces with $\chi > 0$, the plane for genus one surfaces with $\chi = 0$, and the hyperbolic space for high genus surfaces with $\chi < 0$ (see Fig 4).

2.3 Hyperbolic Space Models

One of the anomalies of hyperbolic geometry was the realization that it has no isometric embedding in Euclidean space. Here are three common non-isometric embeddings for hyperbolic geometry. One is the Poincaré model. Others are the Klein model and the Upper Half Plane Model.

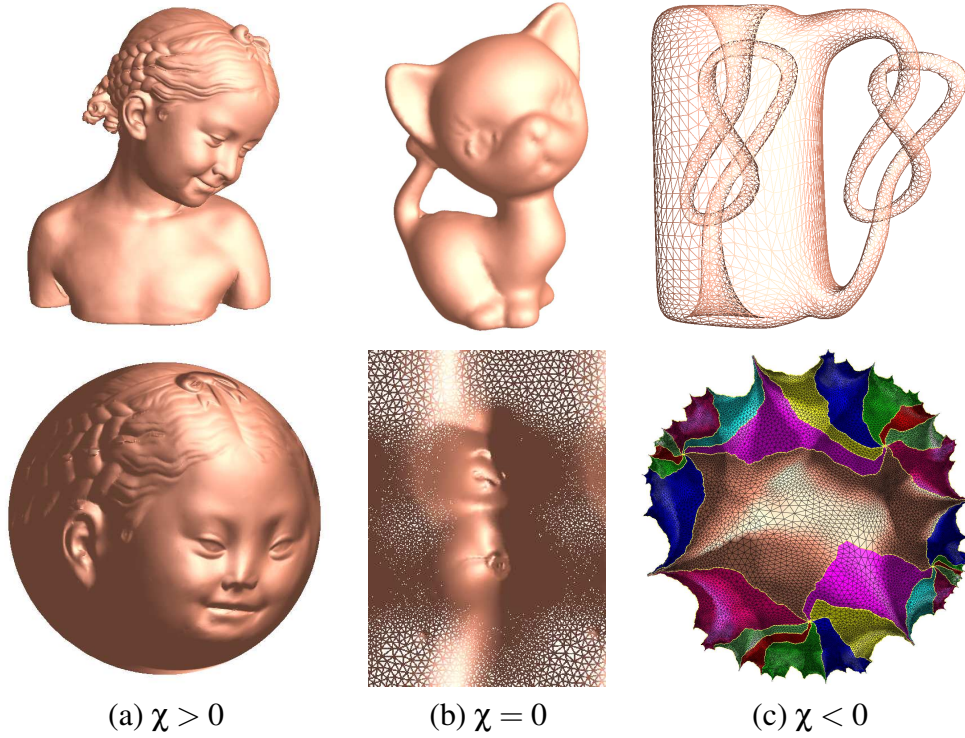


Figure 4: Uniformization Theorem: all surfaces with Riemannian metric can be conformally embedded onto three canonical spaces: sphere, plane and hyperbolic space.

2.3.1 Poincaré Model

The Poincaré model is a unit disk \mathbb{D}^2 in the complex plane with the Riemannian metric $ds^2 = \frac{4dzd\bar{z}}{(1-z\bar{z})^2}$.

The geodesics are circular arcs perpendicular to the boundary of the unit disk $\partial\mathbb{D}^2$. The isometric transformation in this model is the so called Möbius transformation with the form

$$\phi(z) = e^{i\theta} \frac{z - z_0}{1 - \bar{z}_0 z}, z, z_0 \in \mathbb{C}, \theta \in [0, 2\pi).$$

The above Möbius transformation maps z_0 to the center of the disk, and rotates the whole disk by angle θ . Hyperbolic circles are also Euclidean circles.

The Poincaré model is a conformal model, whereas the Klein model is a real projective model.

2.3.2 Klein Model

The Klein model is another model of the hyperbolic space also defined on the unit disk \mathbb{D}^2 . Any geodesic in the Klein model is a chord of the unit circle of the boundary of \mathbb{D}^2 . The map from the Poincaré model to the Klein model is $\beta : \mathbb{H}^2 \rightarrow \mathbb{D}^2$,

$$\beta(z) = \frac{2z}{1 + \bar{z}z}, \beta^{-1}(z) = \frac{1 - \sqrt{1 - \bar{z}z}}{\bar{z}z}z. \quad (6)$$

Any Möbius transformation in the Poincaré model ϕ becomes a real projective transformation in the Klein model $\beta \circ \phi \circ \beta^{-1}$.

2.3.3 Upper Half Plane Model

The upper half plane model takes the Euclidean upper half plane as the "plane". The geodesics are portions of circles with their center on the boundary.

2.4 Fundamental Group and Universal Cover

Two curves are *homotopic* to each other, if they can deform to each other on the surface. Closed loops are classified to homotopy classes by homotopic relation. Two closed curves sharing common points can be concatenated to form another loop. This operation defines the multiplication of homotopic classes. Therefore, all the base pointed homotopy classes form the so called *the first fundamental group* of Σ , and are denoted as $\pi_1(\Sigma)$.

The fundamental group is finitely generated. Suppose a surface Σ is with g handles, and there are two distinct generators a, b on each handle. If they intersect once, but disjoint with generators on other handles, then all g pairs of generators form a set of *canonical fundamental group basis*, denoted as $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$.

Suppose that $\bar{\Sigma}$ and Σ are surfaces, then $(\bar{\Sigma}, \pi)$ is said to be a *covering space* of Σ if π is surjective and locally homeomorphic. Furthermore, if $\bar{\Sigma}$ is simply connected, $(\bar{\Sigma}, \pi)$ is the *universal covering space* of Σ .

A transformation of the universal covering space $\phi : \bar{\Sigma} \rightarrow \bar{\Sigma}$ is a *deck transformation*, if $\pi = \pi \circ \phi$. All deck transformations form a group G . It is also called the *Fuchsian group* of Σ if the transformation is hyperbolic isometry.

The deck transformation group is isomorphic to the fundamental group. Suppose $p \in \Sigma$ is an arbitrary point on Σ , its pre-images are $\pi^{-1}(p) = \{p_0, p_1, p_2, \dots, p_n, \dots\}$ on $\bar{\Sigma}$. Suppose a deck transformation $\phi \in G$ maps p_0 to p_k , then a curve on the universal covering space

$$\gamma: [0, 1] \rightarrow \bar{\Sigma}, \gamma(0) = p_0, \gamma(1) = p_k,$$

connects p_0 and p_k and its projection $\pi(\gamma)$ is a loop on Σ . The homotopy class of $\pi(\gamma)$ is solely determined by p_0 and p_k , independent of the choice of γ . By this way, we get a bijective map from deck transformations to the first fundamental group of Σ .

A *fundamental domain* F is a subset of $\bar{\Sigma}$, such that the universal covering space is the union of conjugates of F , and any two conjugates have no interior point in common. Given a canonical fundamental group generators $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$, we can slice Σ along the curves and get a fundamental domain with boundary $a_1 b_1 a_1^{-1} b_1^{-1} a_2 b_2 a_2^{-1} b_2^{-1} \dots a_g b_g a_g^{-1} b_g^{-1}$.

For any surface Σ , its uniformization metric is also a metric for its universal cover $\bar{\Sigma}$. The universal cover can be isometrically embedded in one of the three canonical spaces: sphere, plane and hyperbolic space.

2.5 Surface Ricci Flow

Suppose S is a smooth surface with Riemannian metric \mathbf{g} . On a local coordinate system, the metric tensor is represented as

$$\mathbf{g} = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix}. \quad (7)$$

The Ricci flow is the process to deform the metric $\mathbf{g}(t)$ according to its induced Gauss curvature $K(t)$, where t is the time parameter

$$\frac{dg_{ij}(t)}{dt} = -K(t)g_{ij}(t). \quad (8)$$

Suppose $T(t)$ is a temperature field on the surface. The heat diffusion equation is

$$\frac{dT(t)}{dt} = -\Delta T(t), \quad (9)$$

where Δ is the Laplace-Beltrami operator induced by the surface metric. The temperature field becomes more and more uniform with the increase of t . Eventually, it will become constant.

It is proven that the curvature evolution induced by the Ricci flow is exactly the same as heat diffusion on the surface

$$\frac{K(t)}{dt} = -\Delta_{\mathbf{g}(t)}K(t), \quad (10)$$

where $\Delta_{\mathbf{g}(t)}$ is the Laplace-Beltrami operator induced by the metric $\mathbf{g}(t)$. It has been proven that Ricci flow converges. The metric $\mathbf{g}(t)$ is conformal to the original metric at any time t . Eventually, the Gauss curvature will become constant just like the heat diffusion $K(\infty) \equiv \text{const}$. The corresponding metric $\mathbf{g}(\infty)$ is called the *uniformization metric*.

Ricci flow can be easily modified to compute a metric with a prescribed curvature \bar{K} , then the flow becomes

$$\frac{dg_{ij}(t)}{dt} = -(\bar{K} - K)g_{ij}(t). \quad (11)$$

If the target curvature \bar{K} is admissible, then the solution metric $\mathbf{g}(\infty)$ induces \bar{K} and is conformal to $\mathbf{g}(0)$.

The following theorems postulate that the Ricci flow defined in 8 is convergent and lead to conformal uniformization metric.

Theorem 2 (Hamilton 1982) *For a closed surface of non-positive Euler characteristic, if the total area of the surface is preserved during the flow, the Ricci flow will converge to a metric such that the Gaussian curvature is constant everywhere.*

Theorem 3 (Chow) *For a closed surface of non-positive Euler characteristic, if the total area of the surface is preserved during the flow, the Ricci flow will converge to a metric such that the Gaussian curvature is constant everywhere.*

Chapter 3

Theoretical Background on Discrete Surfaces

In engineering fields, smooth surfaces are often approximated by simplicial complexes (triangle meshes). Major concepts, such as metrics, curvature, and conformal deformation in the continuous setting can be generalized to the discrete setting. We denote a triangle mesh as Σ , a vertex set as V , an edge set as E , and a face set as F . e_{ij} represents the edge connecting vertices v_i and v_j , and f_{ijk} denotes the face formed by v_i , v_j , and v_k .

Background Geometry In graphics, it is always assumed that a mesh Σ is embedded in the three dimensional Euclidean space \mathbb{R}^3 , and therefore each face is Euclidean. In this case, we say the mesh is with Euclidean background geometry (see Fig. 4(b)). The angles and edge lengths of each face satisfy the Euclidean cosine law.

Similarly, if we assume that a mesh is embedded in the three dimensional sphere \mathbb{S}^2 , then each face is a spherical triangle. We say the mesh is with spherical background geometry (see Fig. 4(a)). The angles and the edge lengths of each face satisfy the spherical cosine law.

Furthermore, if we assume that a mesh is embedded in the three dimensional hyperbolic space \mathbb{H}^2 , then all faces are hyperbolic triangles. We say the mesh is with hyperbolic background geometry (see Fig. 4(c)). The angles and the edge lengths of each face satisfy the hyperbolic cosine law.

In the following discussion, we will explicitly specify the background geometry for a mesh when it is needed. Otherwise, the concept or the algorithm is appropriate for all kinds of the background geometries.

3.1 Discrete Riemannian Metric

A Riemannian metric on a mesh Σ is a piecewise constant metric with cone singularities. A metric on a mesh with Euclidean background metric is a Euclidean metric with cone singularities. Each vertex is a cone singularity. Similarly, a metric on a mesh with spherical (or hyperbolic) background geometry is a spherical (or hyperbolic) metric with cone singularities.

The edge lengths of a mesh Σ are sufficient to define this Riemannian metric,

$$l : E \rightarrow \mathbb{R}^+, \quad (12)$$

as long as, for each face f_{ijk} , the edge lengths satisfy the triangle inequality: $l_{ij} + l_{jk} > l_{ki}$.

3.2 Discrete Gaussian Curvature

The discrete Gaussian curvature K_i on a vertex $v_i \in \Sigma$ can be computed from the angle deficit,

$$K_i = \begin{cases} 2\pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, & v_i \notin \partial\Sigma \\ \pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, & v_i \in \partial\Sigma \end{cases} \quad (13)$$

where θ_i^{jk} represents the corner angle attached to vertex v_i in the face f_{ijk} , and $\partial\Sigma$ represents the boundary of the mesh. The discrete Gaussian curvatures are determined by the discrete metrics.

3.3 Discrete Gauss-Bonnet Theorem

The Gauss-Bonnet theorem 5 states that the total curvature is a topological invariant. It still holds on meshes as follows.

$$\sum_{v_i \in V} K_i + \lambda \sum_{f_i \in F} A_i = 2\pi\chi(M), \quad (14)$$

where A_i denotes the area of face f_i , and λ represents the constant curvature for the background geometry; $+1$ for the spherical geometry, 0 for the Euclidean geometry, and -1 for the hyperbolic geometry.

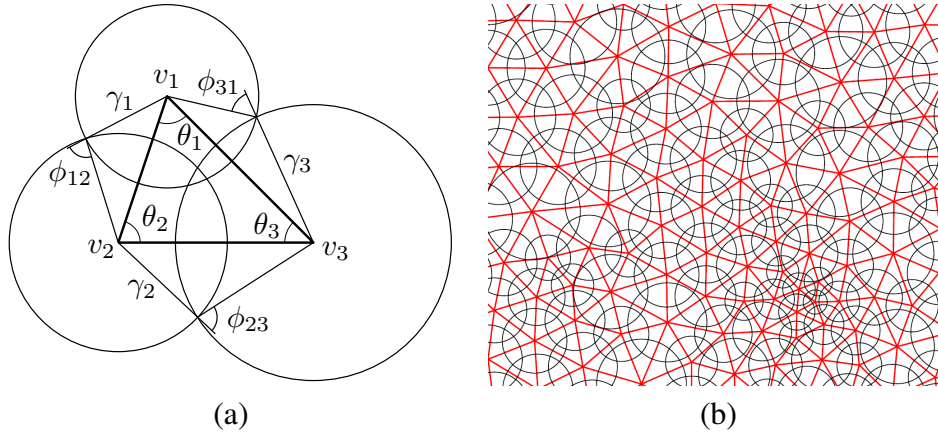


Figure 5: Circle Packing Metric (a) Circle packing metric on one triangle. (b) Circle packing metric on planar triangulations.

3.4 Discrete Conformal Deformation

Conformal metric deformations preserve infinitesimal circles and the intersection angles among them. This fact inspires Thurston to design circle packing metric [48] to approximate discrete conformal metric deformation, which uses circles with finite radii to approximate the infinitesimal circles.

Fig. 5 illustrates the concept of circle packing metric. Each vertex v_i has a circle. For each edge e_{ij} , the intersection angle is defined by the two circles centered on v_i and v_j , which either intersect or are tangent. More precisely, let Γ

be a function defined on the vertices, $\Gamma : V \rightarrow \mathbb{R}^+$, which assigns a radius γ_i to the vertex v_i . Similarly, let Φ be a function defined on the edges, $\Phi : E \rightarrow [0, \frac{\pi}{2}]$, which assigns an acute angle $\Phi(e_{ij})$ to each edge e_{ij} and is called a *weight* function on the edges. The pair of vertex radius function and edge weight function on a mesh Σ , (Γ, Φ) , is called a *circle packing metric* of Σ .

Two circle packing metrics (Γ_1, Φ_1) and (Γ_2, Φ_2) on the same mesh are *conformally equivalent* if $\Phi_1 \equiv \Phi_2$. A *conformal deformation* of a circle packing metric only modifies the vertex radii while preserves the intersection angles on the edges.

3.5 Admissible Curvature Space

A mesh Σ with edge weight Φ is called a *weighted mesh*, which is denoted as (Σ, Φ) . In the following, we want to clarify the spaces of all possible circle packing metrics and all possible curvatures of a weighted mesh.

Let the vertex set be $V = \{v_1, v_2, \dots, v_n\}$, and the radii be $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_n\}$. Let u_i be

$$u_i = \begin{cases} \log \gamma_i & \mathbb{E}^2 \\ \log \tanh \frac{\gamma_i}{2} & \mathbb{H}^2 \\ \log \tan \frac{\gamma_i}{2} & \mathbb{S}^2 \end{cases} \quad (15)$$

where \mathbb{E}^2 , \mathbb{H}^2 , and \mathbb{S}^2 indicate the background geometry of the mesh. We represent a circle packing metric on (Σ, Φ) by a vector $\mathbf{u} = (u_1, u_2, \dots, u_n)^T$. Similarly, we represent the Gaussian curvatures at mesh vertices by the curvature vector $\mathbf{k} = (K_1, K_2, \dots, K_n)^T$. All the possible \mathbf{u} 's form the *admissible metric space*, all the possible \mathbf{k} 's form the *admissible curvature space*.

According to the Gauss-Bonnet theory (Eq. 14), the total curvature must be $2\pi\chi(\Sigma)$, and therefore the curvature space is $n - 1$ dimensional. We add one linear constraint to the metric vector \mathbf{u} , $\sum u_i = 0$, for the normalized metric. As a result, the metric space is also $n - 1$ dimensional. If all the intersection angles are acute, then the edge lengths induced by a circle packing satisfy the triangle inequality. There is no further constraint on \mathbf{u} . Therefore, the admissible metric space is simply \mathbb{R}^{n-1} .

A curvature vector \mathbf{k} is *admissible* if there exists a metric vector \mathbf{u} , which induces \mathbf{k} . The admissible curvature space of a weighted mesh (Σ, Φ) is a convex polytope, specified by the following theorem. The detailed proof can be found in [8].

Theorem 4 Suppose (Σ, Φ) is a weighted mesh with Euclidean background geometry, I is a proper subset of vertices, F_I is the set of faces whose vertices are in I and the link set $Lk(I)$ is formed by faces (e, v) , where e is an edge and v is the third vertex in the face,

$$Lk(I) = \{(e, v) | e \cap I = \emptyset, v \in I\},$$

then a curvature vector \mathbf{k} is admissible if and only if

$$\sum_{v_i \in I} K_i > - \sum_{(e, v) \in Lk(I)} (\pi - \phi(e)) + 2\pi\chi(F_I).$$

The admissible curvature spaces for weighted meshes with hyperbolic or spherical background geometries are more complicated. We refer readers to [24] for detailed discussions.

3.6 Discrete Surface Ricci Flow

Suppose (Σ, Φ) is a weighted mesh with an initial circle packing metric. The discrete Ricci flow is defined as follows.

$$\frac{du_i(t)}{dt} = (\bar{K}_i - K_i), \quad (16)$$

where $\bar{\mathbf{k}} = (\bar{K}_1, \bar{K}_2, \dots, \bar{K}_n)^T$ is the user defined target curvature. Discrete Ricci flow is in the exact same form as the smooth Ricci flow (Eq. 11), which deforms the circle packing metric according to the Gaussian curvature, as in Eq. 16.

Discrete Ricci flow can be formulated in the variational setting, namely, it is a negative gradient flow of a special energy form. Let (Σ, Φ) be a weighted mesh with spherical (Euclidean or hyperbolic) background geometry. For two arbitrary vertices v_i and v_j , the following symmetric relation holds:

$$\frac{\partial K_i}{\partial u_j} = \frac{\partial K_j}{\partial u_i}.$$

Let $\omega = \sum_{i=1}^n K_i du_i$ be a differential one-form [54]. The symmetric relation guarantees that the one-form is closed (curl free) in the metric space.

$$d\omega = \sum_{i,j} \left(\frac{\partial K_i}{\partial u_j} - \frac{\partial K_j}{\partial u_i} \right) du_i \wedge du_j = 0.$$

By Stokes theorem, the following integration is path independent,

$$f(\mathbf{u}) = \int_{\mathbf{u}_0}^{\mathbf{u}} \sum_{i=1}^n (\bar{K}_i - K_i) du_i, \quad (17)$$

where \mathbf{u}_0 is an arbitrary initial metric. Therefore, the above integration is well defined, so called the *discrete Ricci energy*. The discrete Ricci flow is the negative gradient flow of the discrete Ricci energy. The discrete metric which induces $\bar{\mathbf{k}}$ is the minimizer of the energy.

Computing the desired metric with user-defined curvature $\bar{\mathbf{k}}$ is equivalent to minimizing the discrete Ricci energy. For the Euclidean (or hyperbolic) case, the discrete Ricci energy (see Eq. 17) has been proven to be strictly convex (namely, its Hessian is positive definite) in [8]. The global minimum uniquely exists, corresponding to the metric $\bar{\mathbf{u}}$, which induces $\bar{\mathbf{k}}$. The discrete Ricci flow converges to this global minimum.

Theorem 5 (Chow & Luo: Euclidean Ricci Energy) *The Euclidean Ricci energy $f(\mathbf{u})$ on the space of the normalized metric $\sum u_i = 0$ is strictly convex.*

Theorem 6 (Chow & Luo: Hyperbolic Ricci Energy) *The hyperbolic Ricci energy is strictly convex.*

Although the spherical Ricci energy is not strictly convex, the desired metric $\bar{\mathbf{u}}$ is still a critical point of the energy. In our experiments, the solution can be reached using Newton's method.

Chapter 4

Conformal Structure

4.1 Overview

Conformal maps are also called angle preserving maps, and locally distances and areas are only changed by a scaling factor. A conformal mapping is intrinsic to the geometry of a mesh, independent of the resolution of the mesh, and preserves the consistency of the orientation.

Because of these nice properties, one big application in computer graphics, computer vision and medical images for conformal structures on surfaces is parametrization. Surface parameterization is the process of mapping a surface to some domain, usually a planar domain, which converts 3D geometric problems to 2D ones, thereby improving the efficiency and simplifying the computation. Parametrization is important for many graphics applications, for example, texture mapping, remeshing, morphing, and registration. The main challenge is to produce a planar triangulation that best matches the geometry of the 3D mesh, minimizing some measure of *distortion*, for example, in angles or areas.

For the purpose of texture mapping, it is important to reduce the distortion between the parameter domain and the surface. In theory, such kind of a parameterization will preserve the shape locally, and is called conformal parameterization, which can be easily induced from surface conformal structure. Conformal parameterization has been proposed for texture mapping [20, 36, 39], geometry remeshing [1], and visualization [2, 17]. Furthermore, all surfaces can be classified easily by conformal

invariants. A method to compute the conformal invariants for meshes is introduced in [18].

Algorithms of computing conformal parameterizations or conformal structures can be roughly classified into three categories, according to the types of their outputs: *vector-valued mapping functions*, *holomorphic differential forms* and *flat Riemannian metrics*.

Mappings The first category computes a mapping from a given surface to a planar domain by minimizing the harmonic energy of the mapping function. Levy *et al.* [36] approximate the solution to the Cauchy-Riemann equations by finite element method. They compute a piecewise linear conformal mapping from a plane to a surface using least-square energy minimizations. Desbrun *et al.* [9] provide a discrete intrinsic parameterization by minimizing Dirichlet energy. Pinkall and Polthier [43] introduce a discrete version of the Laplace-Beltrami operator, which has been successfully applied for discrete conformal mappings. Based on the mean value theorem in harmonic mappings, Floater [11] provide mean value coordinates, which guarantee flipping-free parameterization. In [9,56], optimal parameterization methods are introduced using free boundaries.

Holomorphic 1-forms Basically, holomorphic 1-forms represent the derivative of a conformal mapping, and they can be considered as vector fields defined on surfaces. Gu and Yau [19] introduce discrete holomorphic 1-forms to compute global conformal surface parameterizations for high genus surfaces. Jin *et al.* [27] compute optimal holomorphic 1-forms to reduce area distortion. Gortler *et al.* [12] use discrete 1-forms to parameterize high genus meshes. Recently, Tong *et al.* [50] generalized the 1-form method to incorporate cone singularities. The main disadvantages of using Holomorphic 1-forms is that it is hard to control the number of singular points and their positions.

Metrics The most general way to compute conformal parameterizations is to compute a flat Riemannian metric which is conformal to the original induced Euclidean metric of the input surface. Much research has been done on the computational algorithms for solving such metrics, including angle based flattening (ABF),

circle packing, circle patterns, and discrete Ricci flows.

Conformal mappings preserve angles. This fact inspires the angle based flattening method [45]. ABF method computes a special configuration of angles of the input mesh, such that the metric is flat and the angle distortions are minimized. The stability and efficiency of ABF method is greatly improved in ABF++ [46] by using advanced numerical method and hierarchical approach.

Kharevych et al. used circle patterns for discrete conformal mappings in [34]. The Euclidean flat cone metric with user prescribed singularities can be obtained by two stage optimizations. Jin et al. used circle packing to design flat cone metrics in [24], which handles spherical, Euclidean and hyperbolic discrete metrics. The algorithm is the discrete analogy of Ricci flow [21]. A linear metric scaling method for computing Euclidean flat cone metric with prescribed curvatures is introduced in [3], where the cone singularities can be automatically selected to minimize the distortion. Circle pattern and discrete Ricci flow are non-linear methods, requires a preprocessing stage, and get an accurate metric; the metric scaling method is linear and flexible for general meshes but with less accuracy.

4.1.1 Applications for Conformal Structure

4.1.1.1 Texture Mapping

Least Squares Conformal Maps for Automatic Texture Atlas Generation In this paper [36], the model to be textured is decomposed into parts with natural shapes, which are homeomorphic to discs, referred to as charts, and each chart is provided with a quasi-conformal parametrization based on a least-squares approximation of the Cauchy-Riemann equations introduced in this paper, then a new packing algorithm is used to gather them in texture space.

Uniform Texture Synthesis and Texture Mapping Using Global Parametrization paper [53] uses the global conformal parameterization to convert the 3D surface texture synthesis problem to a 2D image synthesis problem, which is more intuitive, easier, and conceptually simpler. While the conformality of the parameterization naturally preserves the angles of the texture, they provide a multi-scale technique to maintain a more uniform area scaling factor. This multi-scale method

synthesizes nonuniform textures on a 2D geometry image by considering the area stretching factor (the inverse of the conformal factor) in order to obtain the uniform 3D textures.

4.1.1.2 Remeshing

Interactive Geometry Remeshing Paper [1] introduces an interactive remeshing for surface with irregular geometry. First, the original (arbitrary genus) mesh is substituted by a series of 2D maps in parameter space, including conformal parametric domain, area stretching map, mean curvature map and Gaussian curvature map. The user can easily combine these maps to create a control map which controls the sampling density over the surface patch. This map is then sampled at interactive rates allowing the user to easily design a tailored resampling. Once this sampling is complete, a Delaunay triangulation and fast optimization are performed to perfect the final mesh.

4.1.1.3 Surface Mapping

Genus zero surface conformal mapping and its application to brain surface mapping Paper [17] proposes a method which can find a unique mapping between any two genus zero manifolds by minimizing the harmonic energy of the map with some constraints added to ensure that the conformal map is unique.

In this chapter, we first introduce our optimal global conformal parametrization method using discrete one-form (Section 4.2) [27]. Since the space of holomorphic 1-forms on a closed high genus g surface is g complex dimensional, finding a desired conformal parametrization equals to finding a desired holomorphic 1-form, which can be formulated as a finite dimensional optimization problem under a $2g$ *holomorphic one form* basis space $\Omega(S)$.

Then we present another more powerful tool: discrete Euclidean Ricci flow, which is more flexible for computing high genus surfaces' global conformal parametrizations due to its control of zero points and curvatures on boundaries (Section 4.3) [24].

Several applications based on surface conformal structure are given. One is virtual colon flattening using discrete one-form method to enhance virtual

colonoscopy systems (Section 4.4.1) [23]. The other is N-RoSy field design on general surfaces using discrete Euclidean Ricci flow (Section 4.4.2).

4.2 Optimal Global Conformal Parametrization Using Discrete One-form

All holomorphic 1-forms on a genus g surface with g equal to or greater than one form a linear space, and the basis for such a linear space is $2g$ dimensional. We compute a basis denoted as $\{\omega_1, \omega_2, \dots, \omega_{2g}\}$, such that any linear combination of them is still a holomorphic 1-form. By integrating a holomorphic 1-form, a global conformal parametrization can be obtained. We formulate different energies to measure the quality of the global conformal parametrization. One is to measure the uniformity of the parametrization and the other is to measure the ratio of parameter area on regions of interest in the surface. They are both denoted as $E(\omega)$, with $\omega = \sum_{i=1}^{2g} \lambda_i \omega_i$, and $E(\omega) = E(\lambda_1, \lambda_2, \dots, \lambda_{2g})$. We need to find the linear combination coefficients λ_i to optimize the energy. The necessary condition for the optimal holomorphic one form is straight forward,

$$\frac{\partial E}{\partial \lambda_i} = 0, i = 1, 2, \dots, 2g.$$

If the Hessian matrix $(\frac{\partial^2 E}{\partial \lambda_i \partial \lambda_j})$ is positive definite, E will reach the minimum; if the Hessian matrix is negative definite, E will be maximized.

For closed genus zero surfaces, there is no holomorphic 1-form. The global conformal parametrization is a conformal map $\phi : S \rightarrow S^2$ from the surface S to the unit sphere S^2 . Two such kinds of transformations differ by a Möbius transformation on S^2 . Suppose both ϕ_1 and ϕ_2 are two conformal parameterizations of S , consequently

$$\phi_2 \circ \phi_1^{-1} = \mu,$$

where μ is a Möbius automorphism of the sphere. All conformal maps from S to S^2 can be formulated as $\mu \circ \phi_1$. We compute one conformal map ϕ_1 first, then compose it with a Möbius transformation μ . By choosing appropriate μ , we can optimize the energy.

For open genus zero surfaces, it can be globally conformally parameterized by the unit disk. Two such kinds of parameterizations differ by a Möbius transformation defined on the disk. We can find the best one using a similar method.

Topology modification is necessary for two purposes. First, we will slice the surface open along feature lines and the feature lines will be parameterized as the boundaries in parameter domain. When two surfaces are matched, the feature lines can be matched automatically by matching boundaries on the parameter domain. In other words, we convert geometric matching to topology matching. Second, in order to improve the uniformity of the parametrization, we need to modify the topology. For surface regions like narrow tubes, the stretching factor will increase exponentially with respect to the height of the tube. In order to improve the uniformity of the parametrization, we need to modify the topology by introducing small boundaries at the top of the tube. We introduce an algorithm to automatically locate such extreme points and modify the topology.

For genus g surfaces, there will be $2g - 2$ zero points on any global conformal parametrization. The stretching factor of a zero point is zero. The neighborhood of the zero point will be mapped to a very small area on the parameter plane. For purpose of sampling the neighborhood will be under-sampled. For texture mapping purpose, zero points will cause irregular patterns. Therefore, it is desirable to hide the zero points at the predetermined positions. The distribution of zero points are determined by the conformal structure of the surface. Zero points can not be allocated arbitrarily. However, we can still control part of the zero points.

4.2.1 Optimal Global Conformal Parameterizations of High Genus Surfaces

For high genus surfaces, their conformal structures can be represented as a holomorphic 1-form basis, which are $2g$ functions $\omega_i : K_1 \rightarrow \mathbb{R}^2, i = 1, 2, \dots, 2g$. Any holomorphic 1-form ω is a linear combination of these functions. The surface can be cut open to a topological disk, namely a *fundamental domain*. By integrating ω on a fundamental domain, the whole surface can be globally conformally mapped to the uv plane.

The algorithm to compute the holomorphic 1-form basis $\{\omega_i, i = 1, 2, \dots, 2g\}$

for a triangular mesh is as follows:

- Compute the first homology basis of this triangular mesh.
- Compute the first cohomology basis of this triangular mesh.
- Compute harmonic 1-form basis from computed cohomology basis in last step using heat flow method.
- For each harmonic 1-form basis, locally rotate it by a right angle about the normal to get a new one, and the two form a holomorphic 1-form basis.

The computation process is equivalent to solving an elliptic partial differential equation on the surface using finite element method. The details for computing holomorphic 1-form are thoroughly explained in [18] and [19].

If surface is with high genus, all its holomorphic one forms form a linear space. We formulate the functionals as polynomials of the coefficients of $\{\lambda_1, \lambda_2, \dots, \lambda_{2g}\}$. Also, we set the total area in the parameter domain to be fixed. Then we use the Lagrange multiplier and Newton's method to optimize it.

Uniform Global Conformal Parametrization Given any holomorphic one-form ω , $\omega = \sum_{k=1}^{2g} \lambda_k \omega_k$, we require the total parameter area equals to the total area of the surface in R^3 . The constraint functional is:

$$\sum_{[u,v,w] \in K_2} \frac{1}{2} |\omega([u,v]) \times \omega([v,w])| = \sum_{[u,v,w] \in K_2} S_{[u,v,w]}, \quad (18)$$

where $S_{[u,v,w]}$ is the area of face $[u,v,w]$ in R^3 . The uniformity functional is defined as the sum of the squared area differences of faces between parameter area and area in R^3 ,

$$E(\omega) = \sum_{[u,v,w] \in K_2} \left(\frac{1}{2} |\omega([u,v]) \times \omega([v,w])| - S_{[u,v,w]} \right)^2. \quad (19)$$

Both the constraint and the energy functional are polynomials with respect to λ_i 's. For example, the constraint can be reformulated as a quadratic form as $\sum_{i,j=1}^{2g} c_{ij} \lambda_i \lambda_j = \text{const}$ with $c_{i,j} = \sum_{[u,v,w] \in K_2} \frac{1}{2} |\omega_i([u,v]) \times \omega_j([v,w])|$.

We use the Newton's method to optimize the uniformity energy with the constraint. Because the energy is quintic, the extremal points are not unique. We randomly set initial values for λ_i 's, by minimizing the uniformity energy, we get

the most uniform parametrization; by maximizing the uniformity energy, we get the least uniform parametrization.

Emphasized Global Conformal Parametrization It is also desirable to allocate more parameter areas for special regions on the surface in real applications. For example, in surface remeshing, more samples are required for regions with high Gaussian curvature or sharp features. We design another functional to measure the quality of parametrization for emphasized regions.

Suppose we subdivide the whole surface to two regions D_0 and D_1 . They two may be disconnected, with complicated topologies. If we want to maximize the parameter areas for D_0 , we define the emphasized area energy as,

$$E(\omega) = \frac{1}{2} \sum_{[u,v,w] \in D_0} |\omega([u,v]) \times \omega([v,w])|. \quad (20)$$

with the same constraint in equation 18.

The functional can be represented as a quadratic form directly. Let $c_{i,j} = \sum_{[u,v,w] \in D_0} |\omega_i([u,v]) \times \omega_j([v,w])|$, then the emphasized area energy is

$$E(\lambda_1, \lambda_2, \dots, \lambda_{2g}) = \sum_{i,j=1}^{2g} c_{ij} \lambda_i \lambda_j. \quad (21)$$

We use conjugate gradient method for the optimization after setting initial values for λ_i 's. By maximizing this functional, we put more samples on D_0 ; By minimizing it, we put more samples on D_1 .

Figure 13 is the result of area emphasized optimization for human body model. Figure 13 (b), (c) and (d) emphasize the bottom part, the left part, and the right part, with the ration of the parameter area 82.1%, 88.6%, and 87.3% over the total parameter area respectively.

4.2.2 Optimal Global Conformal Parameterizations of Genus Zero Surfaces

For closed genus 0 surfaces, they can be conformally parameterized by a unit sphere. For genus 0 surfaces, Harmonic maps are equivalent to conformal maps. We



Figure 6: Max Planck Head Model. (a).Most uniform conformal parametrization, energy: $1.605e-5$ (b). Least uniform conformal parametrization, energy: $4.231e-5$

use Gauss map as the initial map, then the heat flow method to reduce the harmonic energy with special constraints. The final harmonic map is a global conformal parametrization. By composing a Möbius map of the sphere, we can obtain all possible global conformal parameterizations.

For open genus 0 surfaces, we double cover it and get a closed symmetric surface. We can map this double covered surface conformally to the sphere and preserve the symmetry with each copy of the original surface mapped to one hemisphere. After we use stereo-graphic projection map one hemisphere to the unit disk, the surface is then globally conformally parameterized by the disk. By composing a Möbius map of the disk, we can construct all global conformal parameterizations for the surface.

For genus zero surfaces, We conformally map them to a unit sphere or a unit disk. Because the parameter domains are fixed, the constraint 18 is unnecessary.

Although they have no holomorphic 1-forms, we can still use the uniformity energy and the emphasized area energy, but the admissible transformations are changed to the Möbius transformations.

Topological sphere A sphere can be conformally mapped to the complex plane by a stereographic projection $\tau : S^2 \rightarrow \mathbb{C}$,

$$\tau(x, y, z) = \frac{x}{1-z} + \sqrt{-1} \frac{y}{1-z}.$$

The Möbius transformation on the complex plane has the formulae

$$\mu(z) = \frac{az+b}{cz+d}, ad-bc=1, a, b, c, d \in \mathbb{C}.$$

A conformal automorphism ϕ of the sphere can be formulated as $\phi = \tau^{-1} \circ \mu \circ \tau$,

Given a topological sphere surface, we first compute its conformal map $\phi_0 : S \rightarrow S^2$ to the sphere. All its admissible conformal mappings can be represented as $\phi_\mu = \tau^{-1} \circ \mu \circ \tau \circ \phi_0$.

The uniformity functional becomes

$$E(\mu) = \sum_{[u,v,w] \in K_2} (|\phi_\mu(r(u)), \phi_\mu(r(v)), \phi_\mu(r(w))| - |r(u), r(v), r(w)|)^2, \quad (22)$$

where $|a, b, c|$ represents the area of the triangle formed by a, b, c . This is a rational formula with respect to the coefficients of μ . We use Newton's method to optimize it without constraint.

Similarly, the emphasized area energy is formulated by

$$E(\mu) = \sum_{[u,v,w] \in D_0} |\phi_\mu(r(u)), \phi_\mu(r(v)), \phi_\mu(r(w))|. \quad (23)$$

We use Newton method to maximize the energy also. Because the optimal solutions are not unique, we randomly choose the initial Möbius transformation μ_0 , and use ϕ_{μ_0} as the initial parametrization.

Topological disk For the topological disk case, we use double covering to make it a symmetric topological sphere. But we restrict the admissible transformations to be in a subgroup of the Möbius group, which preserves the symmetry; namely $\mu(\bar{z}) = \overline{\mu(z)}$.

The formula for such a Möbius transformation can be written as

$$\mu(z) = \frac{az+b}{\bar{b}z+\bar{a}}, a\bar{a}-b\bar{b}=1, a, b \in \mathbb{C}.$$

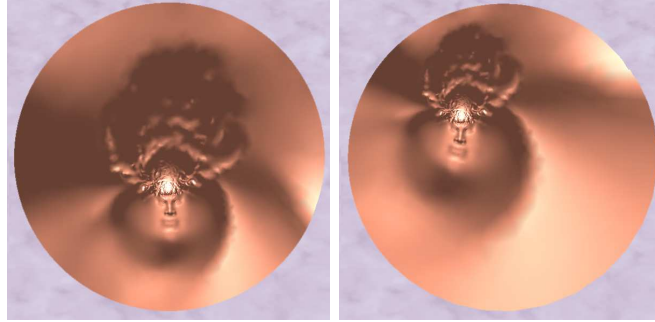


Figure 7: David Head Model. Möbius transformation from the unit disk to itself.

The other steps are similar to those for the case of a topological sphere. The Max Planck head surface in figure 6 is a topological disk. The uniformity energy is minimized by using Möbius transformations. The most uniform global conformal parametrization is illustrated in (a), and the least one is in (b). Figure 7 illustrates a Möbius transformation from the disk to itself.

4.2.3 Topological Optimization

For long and narrow surface regions, such as fingers and tails, the area distortion is huge with conformal parametrization. We will show that for such shapes, we can not get a uniform conformal parametrization by linear combination of the holomorphic one-form basis. We have to change the conformal structure of the surface, which means we need to modify the topology.

First we proof that for long tube shapes, the conformal factor will increase exponentially. Suppose we have a long thin cylinder and we plan to conformally parameterize it. If we use polar coordinates (ρ, θ) with The center of the top mapped to the origin, the conformal factor is a function dependent only on ρ because of symmetry. The Gaussian curvature of the cylinder is zero, and

$$k(\rho, \theta) = \frac{1}{\lambda^2} \Delta \log \lambda = 0. \quad (24)$$

We can deduce $\lambda(\rho) = e^{a\rho+b}$, where a, b are constants. No matter what kind of conformal parametrization we choose, the stretching is exponential, as illustrated by Figure 8 (a), which has two cylinder shaped branches where parameterizations



Figure 8: Cactus Model. (a).Most uniform parametrization with cactus model (b). Most uniform parametrization after one slice on the top of the left branch (c). Most uniform parametrization after another slice on the top of the right branch.

are under sampled. We have to change the topology of the surface by introducing a small boundary at the top of the cylinder and let the conformal factor become constant. Figure 8 (b) shows the new parametrization after one slice on the top of its left branch, and (c) is the final after adding one more slice on the top of the right branch.



Figure 9: Bunny Model With Topological Optimization. (a).Conformal Factor is color encoded into bunny (b). Topological Optimization: three slices in bunny (c). Most Uniform Global Conformal Parametrization, energy: $0.37e-4$ (d). Least Uniform Global Conformal Parametrization, energy: $2.1e-4$

Base on this observation, we design our greedy topological modification algorithm as follows. We first find the most uniform conformal parametrization for

current surface; then we locate points with extremely high conformal factors and introduce small slices at the neighborhoods of those points; After double covering, we recompute its conformal structure. The whole process repeats until the uniformity energy is less than some threshold.

Estimating Conformal Factor Suppose we obtain a global conformal parametrization induced by a holomorphic 1-form ω . The conformal factor for each vertex can be estimated by the following formulae:

$$\lambda(v) = \frac{1}{n} \sum_{[u,v] \in K_1} \frac{|r(u) - r(v)|^2}{|\omega([u,v])|^2}, u, v \in K_0, \quad (25)$$

where n is the valence of vertex v . In practice, we compute $\frac{1}{\lambda}$ instead of computing itself. Then at the extreme points, the inverse of the conformal factor is very close to zero.

Locating the Extreme Points We locate the cluster of vertices with relatively small reversed conformal factors and compute its center of gravity, finding the closest vertex to it. This vertex is an extreme point. Then we introduce a small slice through these extreme points, double cover the surface, and compute a holomorphic 1-form basis. The optimal parametrization of current topology is computed by minimizing the uniformity energy. We repeat the whole procedure until the energy is smaller than the threshold or converges to a limit.

We need to address the question of whether the uniformity would really be improved by this procedure. Suppose at step n , we get a surface S_n . Then any global conformal parameterizations for S_n is also a global conformal parametrization for S_{n+1} , and the minimal uniformity energy of S_{n+1} denoted as E_{n+1} is no greater than that of S_n . The sequence $\{E_0, E_1, E_{n+1}, \dots\}$ is nonincreasing and will converge to a limit. In practice, if the optimal uniformity energy doesn't decrease too much, the procedure will terminate.

The procedure and results for topological optimizing bunny model are illustrated in figure 9. In (a), the bunny is conformally mapped to a sphere. The conformal factors are color encoded with red color indicating high conformal factor regions. Then in (b) we introduce small boundaries on high conformal areas, two

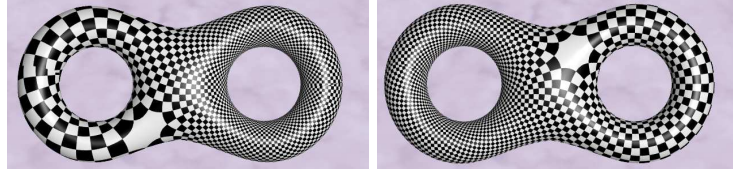


Figure 10: Genus two eight model with zero points locating at different positions.

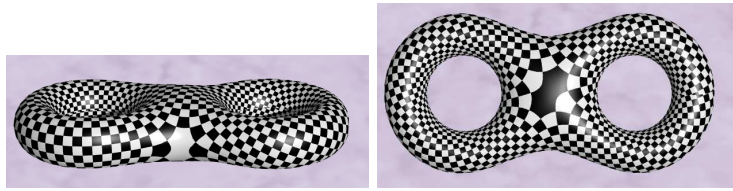


Figure 11: Genus two eight model with zero points locating at different positions.

on the tips of ears and one on the tail, and recompute conformal structure after double covering. (c) shows the best parametrization after the topology modification, and (d) is the worst one with the same topology.

For the horse model and the hull model in figure 15, they all have been introduced 6 boundaries for topological optimization. And the double covering of each model is a genus 5 surface. Their most uniform and least uniform parametrization results are showed respectively.

4.2.4 Zero Points Allocation

Definition 7 (Zero Point). Given a Riemann surface S with a conformal structure A , a holomorphic 1-form ω , with $\omega = f(z)dz$, where $f(z)$ is an analytic function and $z = u + iv$ is the local parameter. If $f(z)$ equals zero at point p , p is a zero point of ω .

For a Riemann surface S with genus g , a holomorphic 1-form ω has $2g - 2$ zero points in principle. The stretching factor of a zero point is zero, and the neighborhood of the zero point will be mapped to a very small area on the parameter plane, which will cause the neighborhood under-sampled, see figure 10 and 11 For texture mapping purpose, zero points will cause irregular patterns. Therefore, it is desirable to allocate zero points at the predetermined less important positions.

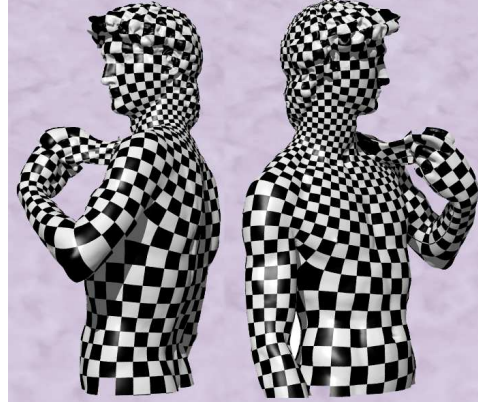
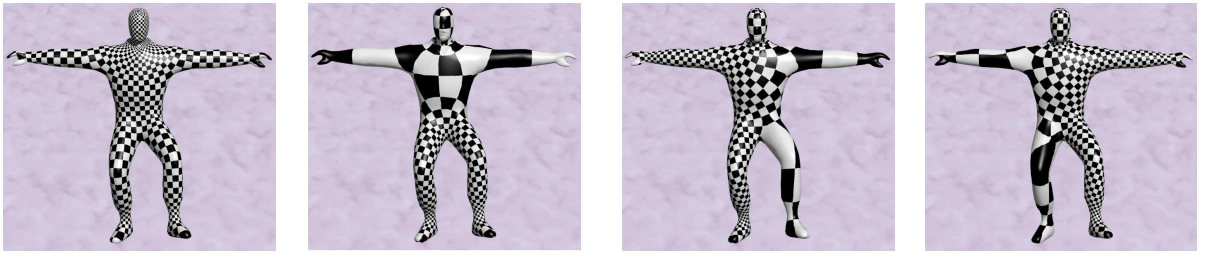


Figure 12: David Half Body Model With Zero Points Hidden. (a). One zero point hidden under left armpit (b). The other zero point hidden under right armpit



a. Most Uniform b. Emphasize Bottom Part. Emphasize Left Part. Emphasize Right Part

Figure 13: Human Body Model.

Although the positions of the zero points are globally related, which means they are determined by the conformal structure of the surface, and impossible to allocate all of them arbitrarily, we can still control part of them.

Suppose ω is a holomorphic 1-form, it has $p_1, p_2, \dots, p_{2g-2}$ zero points, then $\omega(p_i) = 0, \forall i$. Let $\omega = \sum_{j=1}^{2g} \lambda_j \omega_j$, we get the linear system

$$\sum_{j=1}^{2g} \lambda_j \omega_j(p_i) = 0, j = 1, 2, \dots, 2g - 2. \quad (26)$$

If $\{p_1, p_2, \dots, p_{2g-2}\}$ is a set of zero points for some holomorphic 1-form $\omega \neq 0$, it is necessary and sufficient that the matrix $(\omega_j(p_i))$ is degenerated.

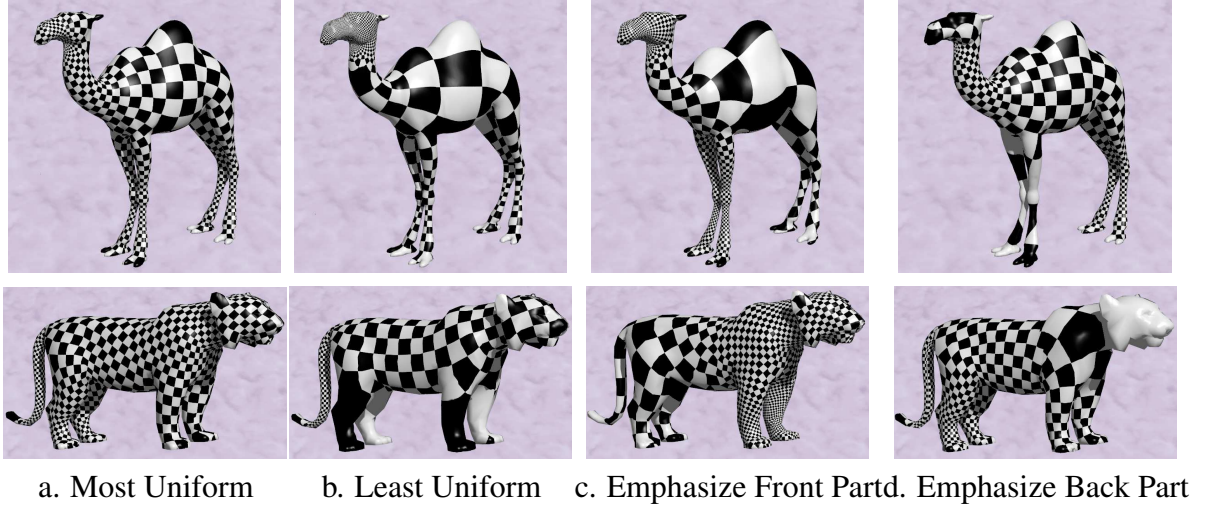


Figure 14: Camel Model and Lion Model.

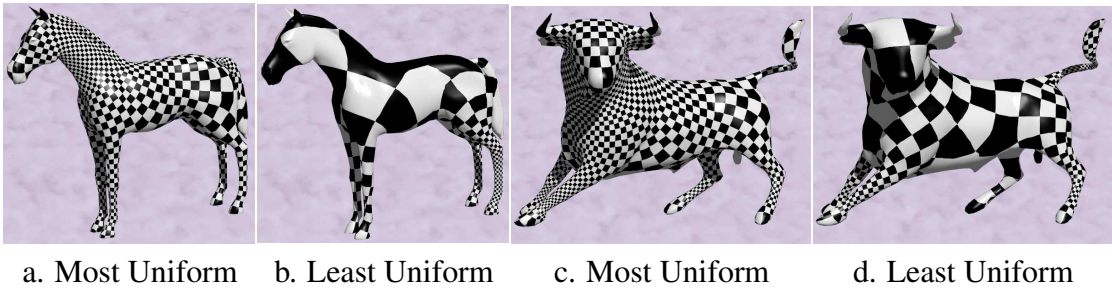


Figure 15: Horse Model and Hull Model.

In our discrete setting, $\omega = \sum_{i=1}^{2g} \lambda_i \omega_i$, and we use the following to approximate $\omega(v), v \in K_0$.

$$\omega(v) = \sum_{[u,v] \in K_1} \frac{\omega([u,v])}{|r(u) - r(v)|} = \sum_{[u,v] \in K_1} \sum_{i=1}^{2g} \frac{\omega_i([u,v])}{|r(u) - r(v)|}.$$

Suppose we want to set n zero points $\{v_1, v_2, \dots, v_n\}$, where $n < 2g - 2$, then

we need to minimize the following energy

$$E(\omega) = \sum_{i=1}^n |\omega(v_i)|^2. \quad (27)$$

This functional is a quadratic form of $\lambda_1, \lambda_2, \dots, \lambda_{2g}$ and can be solved easily using conjugate gradient method. As long as n is not greater than g , we can fix the zero points to the predetermined positions.

For david half body model in Figure 12, We predetermine the positions of two zero points under armpit, and by minimizing the energy 27 to get the desired holomorphic 1-form and parametrization.

4.3 Surface Global Conformal Parameterization Using Discrete Euclidean Ricci Flow

Discrete Euclidean Ricci flow is a more powerful and flexible tool to compute surfaces' global conformal parameterizations. For surfaces with zero Euler number, like the torii and the annuluses, we can find a special metric called flat uniformization metric, which is conformal to the original one, using discrete Euclidean Ricci flow with all target curvatures of vertices set to zero. Then surfaces with their flat uniformization metric can be conformally flattened to the plane.

For surfaces with with non-zero Euler number, according to Gauss-Bonnet theorem, there must be some singularities for the parameterizations where the curvatures are not zeros. We can either concentrate all curvatures onto the singularities, while the target curvatures of all other vertices are set to zero; or if the surface is open, we can push those curvatures to the boundaries and set the target curvatures of interior vertices zero. It is flexible for the numbers and locations of singularities, same as the boundary curvatures of open surfaces, as long as the sum of curvatures satisfy Gauss-Bonnet theorem.

The main issues are the computation of the flat uniformization metric (Section 4.3.1), and the flatten of the surface on plane with such metric (Section 4.3.2).

4.3.1 Computing Flat Uniformization Metric

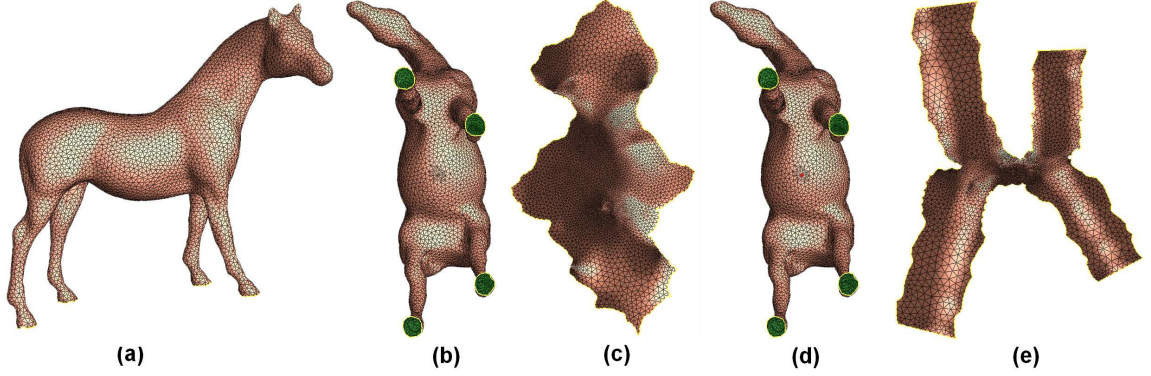


Figure 16: (a) Horse model with four boundaries on its hoofs. (b) All curvatures are pushing to its four boundaries, and all interior vertices are 'flat', with zero Gaussian curvature. (c) The horse is embedded on the plane with the new metric. (d) All curvatures are concentrated on one singularity vertex, which is marked with red. (e) The horse is embedded on the plane with the new metric.

Initial Circle Packing Metric First, we compute an initial circle packing metric (Γ, Φ) which approximates the original Euclidean metric on the given mesh by the following procedures:

1. For each corner of a vertex v_i related with the face f_{ijk} , we compute a radius for the vertex v_i with respect to f_{ijk} :

$$\gamma_i^{jk} = \frac{l_{ki} + l_{ij} - l_{jk}}{2},$$

where l_{ij}, l_{jk}, l_{ki} indicate the lengths of the edges e_{ij}, e_{jk}, e_{ki} , respectively.

2. For each vertex v_i , we approximate the radius γ_i by averaging the radii from the faces adjacent to v_i :

$$\gamma_i = \frac{1}{m} \sum_{f_{ijk} \in F} \gamma_i^{jk},$$

where m is the number of the adjacent faces to vertex v_i .

3. For each edge e_{ij} , we compute its edge weight ϕ_{ij} using the Euclidean cosine law:

$$l_{ij}^2 = \gamma_i^2 + \gamma_j^2 + 2\gamma_i\gamma_j \cos \phi_{ij}, \quad \mathbb{E}^2 \quad (28)$$

where l_{ij} represents the edge length between vertex v_i and v_j .

Discrete Euclidean Ricci Flow With the initial circle packing metric (Γ, Φ) , the discrete Euclidean Ricci flow can be numerically implemented with the following steepest descent method. Notice that during the computation the vertex radii Γ vary with the change of the time while the edge weights Φ are fixed.

1. For the given mesh, assign the radius γ_i for each vertex v_i and the weight ϕ_{ij} for each edge e_{ij} by the initial circle packing metric (Γ, Φ) (see Section 4.3.1).
2. Compute each edge length l_{ij} from current vertices radii γ_i and γ_j and fixed edge weight Φ_{ij} , by using the Euclidean cosine law Eq. 28.
3. Compute the angle θ_i^{jk} related to each corner $i \angle_j^k$, from current edge lengths l_{ij} , l_{jk} and l_{ki} , by using the inverse cosine law:

$$l_{ij}^2 = l_{jk}^2 + l_{ki}^2 - 2l_{jk}l_{ki} \cos \theta_k, \quad \mathbb{E}^2 \quad (29)$$

4. Compute the discrete Gaussian curvature K_i of each vertex v_i by using Equation 13.
5. Update the radius γ_i of each vertex v_i with

$$\gamma_i = \begin{cases} \gamma_i + \varepsilon(\bar{K}_i - K_i)\gamma_i, & \mathbb{E}^2 \end{cases} \quad (30)$$

where \bar{K} is the target Gaussian curvature, the sum under the constraint of the Gauss-Bonnet theorem.

6. The vertex radii need to be normalized such that the sums of all γ_i are equal to 1 or some constant number.
7. Continue the procedures from 2 to 6, until the $\max |\bar{K}_i - K_i|$ among all vertices satisfies the user-specified error tolerance, which in our experiments is set to $1e-6$, which is good enough for later embedding, no folding or overlapping.

Newton's Method Discrete Euclidean Ricci flow is the gradient flow for minimizing the Ricci energy (i.e., Eq. 17), with the convergence and uniqueness guaranteed by Theorem 5. So we can further improve the speed by using Newton's method.

The key step is to compute the Hessian matrix. Let $u_i = \log \gamma_i$, then the Ricci energy (i.e., Eq. 17) is a convex function of u_i . We can measure the variation of the Gaussian curvatures K_i of a given vertex v_i , with respect to the variation of the radii related with v_i and its neighbor vertex v_j (i.e., $\partial K_i / \partial u_i$ and $\partial K_i / \partial u_j$).

From Eqs. 13 and 28 we can obtain the relationship between the Gaussian curvature K_i , γ_i , γ_j , and γ_k as follows.

$$K_i = 2\pi - \sum_{f_{ijk} \in F} \cos^{-1} (\gamma_i^2 + \gamma_i \gamma_j \cos \phi_{ij} + \gamma_k \gamma_i \cos \phi_{ki} - \gamma_j \gamma_k \cos \phi_{jk})$$

From the above equation, we can deduce $\partial K_i / \partial u_i$ and $\partial K_i / \partial u_j$, as follows.

$$\begin{aligned} \frac{\partial K_i}{\partial u_i} &= \frac{\partial \gamma_i}{\partial u_i} \frac{\partial K_i}{\partial \gamma_i} = \gamma_i \sum_{f_{ijk} \in F} \frac{AD - BC}{A\sqrt{A^2 - B^2}} \\ \frac{\partial K_i}{\partial u_j} &= \frac{\partial \gamma_j}{\partial u_j} \frac{\partial K_i}{\partial \gamma_j} = \gamma_j \sum_{f_{ijk} \in F} \frac{AF - BE}{A\sqrt{A^2 - B^2}} \end{aligned}$$

, where

$$\begin{aligned} A &= 2l_{ij}l_{ki} \\ B &= l_{ij}^2 + l_{ki}^2 - l_{jk}^2 \\ C &= 2(\gamma_i + \gamma_j \cos \phi_{ij}) \frac{l_{ki}}{l_{ij}} + 2(\gamma_i + \gamma_k \cos \phi_{ki}) \frac{l_{ij}}{l_{ki}} \\ D &= 2(2\gamma_i + \gamma_j \cos \phi_{ij} + \gamma_k \cos \phi_{ki}) \\ E &= 2(\gamma_j + \gamma_i \cos \phi_{ij}) \frac{l_{ki}}{l_{ij}} \\ F &= 2(\gamma_i \cos \phi_{ij} - \gamma_k \cos \phi_{jk}) \end{aligned}$$

4.3.2 Embedding One Domain on Plane

After the discrete Euclidean ricci flow finishes, we can derive the desired conformal metric represented as the deformed edge lengths from the final circle packing metric, then embed the given mesh onto plane.

4.3.2.1 Computing Edge Lengths

It is straightforward to compute edge lengths e_{ij} from the final circle packing metric: vertices radii γ_i and γ_j , and fixed edge weight ϕ_{ij} , using the Euclidean cosine law Eq. 28.

4.3.2.2 Embedding With Computed Metric

After computing the desired circle packing metric (Γ, Φ) , we embed the mesh to the plane, denoting the embedding map is $\tau : \Sigma \rightarrow \mathbb{R}^2$.

1. Select an arbitrary face f_{012} as the first face to embed, compute the corner angles $\theta_0, \theta_1, \theta_2$. Set $\tau(v_0)$ to be $(0,0)$, $\tau(v_1)$ to be $(l_{01}, 0)$, and $\tau(v_2)$ to be $l_{02}(\cos \theta_0, \sin \theta_0)$. Put all faces sharing an edge with f_{012} to a queue.
2. Pop the first face f_{ijk} from the queue. If all vertices v_i, v_j, v_k have been embedded, continue. Otherwise, assume v_k has not been embedded, then v_i, v_j must have been embedded already. Then the $\tau(v_k)$ is one of the two intersection points of the two circles $|\tau(v_k) - \tau(v_i)| = l_{ki}$ and $|\tau(v_k) - \tau(v_j)| = l_{jk}$. Further more $(\tau(v_j) - \tau(v_i)) \times (\tau(v_k) - \tau(v_i)) > 0$. Put all faces which have not been embedded and share one edge with f_{ijk} to the queue.
3. Repeat step 2 until the queue is empty.

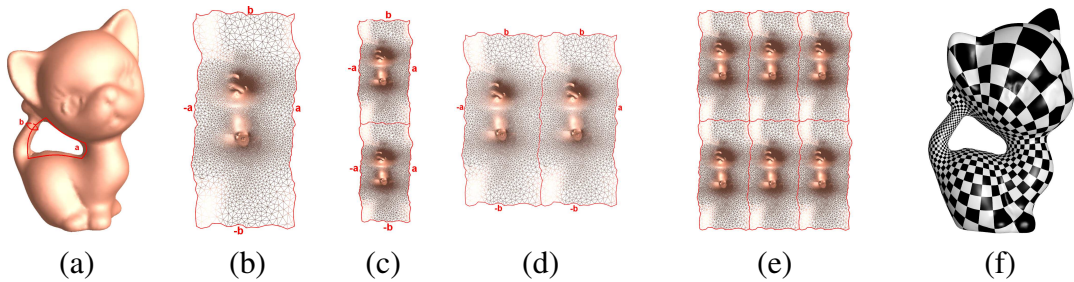


Figure 17: (a) Genus one kitten model marked with a set of canonical fundamental group generators which cut surface into a topological disk with four sides: a, b, a^{-1}, b^{-1} . (b) One period is conformally Embedded onto plane, marked with four sides. (c) One translation moves the side b of one copy of the period matching the side b^{-1} of itself. (d) Another translation moves the side a of another copy of the period matching the side a^{-1} of itself. (e) A portion of tessellate of the kitten in planar domain. (f) The Euclidean coordinate of each vertex in (b) is used as texture coordinate for the kitten model in (a), with check board texture.

4.3.3 Embedding Periodically

A mesh Σ with zero Euler number can be periodically embedded onto the Euclidean plane, which means a layout of infinite copies of the given mesh. To do

tiling, we need to compute the embedding of the canonical fundamental domain of the given mesh first, then generate a set of deck transformations, which will produce the Tessellation. The following will give a detailed explanation.

Embedding the Canonical Fundamental Domain We first compute a set of canonical fundamental group generators $\{a_1, b_1\}$. From definition, a_1 and b_1 are closed loops and with only one geometric intersection point. The algorithms to compute the canonical fundamental group generators have been studied in computational topology and computer graphics literature. We adopted the methods introduced in [6] directly because of its simplicity. Figure 17(a) shows one set of canonical fundamental group generators marked on kitten model with zero Euler number.

If we slice surface M along the curves, we can get a topological disk domain with boundary $\partial M = aba^{-1}b^{-1}$, called canonical fundamental domain. The embedding of the canonical fundamental domain is simple, exactly the same procedure as we introduced in Sec. 4.3.2. Figure 17(b) gives the embedding of the canonical fundamental domain for kitten model on plane.

Computing Deck Transformation Group Generators The embedding of the canonical fundamental domain for a closed zero Euler number surface has 4 different sides, $\rho(a), \rho(b), \rho(a^{-1}), \rho(b^{-1})$ (see Fig. 17(b)), where a coincides with a^{-1} in original model, same as b and b^{-1} , which induce 4 translations defined on the plane. These translations and their compositions form a group called deck transformation group, which can generate the tessellation of the surface on plane.

One deck transformation group generator $\beta : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ which maps the side $\rho(b^{-1})$ of a copy of the fundamental domain $\rho(D)$ coinciding with the side $\rho(b)$ of $\rho(D)$, as shown in Fig. 17(c), can be easily found by computing a translation which moves one randomly chosen boundary edge on side $\rho(b)$ to its mate on side $\rho(a)^{-1}$.

Another deck transformation group generator α which satisfies $\alpha \circ \rho(a) = \rho(a^{-1})$, as shown in Fig. 17(d), can be found similarly,

Generating Tessellation With the computed 2 generators, it is easy to generate more from its compositions. Putting each of these transformations on the canonical

fundamental domain of given surface, we can generate a Tessellation of the surface on its corresponding domain. Figure 17(e) illustrates the tessellation of kitten model on plane.

4.4 Applications

4.4.1 Conformal Virtual Colon Flattening

Virtual colonoscopy uses computed tomographic (CT) images of patient's abdomen and a virtual fly-through visualization system [37] that allows the physician to navigate within a 3D model of the colon searching for polyps, the precursors of cancer. Virtual colonoscopy has been successfully demonstrated to be more convenient and efficient than the real optical colonoscopy. However, because of the length of the colon, inspecting the entire colon wall is time consuming, and prone to errors. Moreover, polyps behind folds may be hidden, which results in incomplete examinations.

In paper [23], we propose a novel method for colon flattening by computing the conformal structure of the surface, represented as a set of holomorphic 1-form basis. It has the following advantages:

1. The algorithm is rigorous and theoretically solid, which is based on the Riemann surface theory and differential geometry;
2. It is general, so it can handle high genus surfaces;
3. The global distortion from the colon surface to the parametric rectangle is minimized, which is measured by harmonic energy;
4. It is angle preserving, so the shape of the polyps is preserved;
5. The topology noise is removed automatically by our shortest loop algorithm. Combined with the direct volume rendering method, the flattened 2D colon image provides an efficient way to enhance virtual colonoscopy systems.

The colon wall is first segmented and extracted from the CT data set. The topology noise (i.e., minute handle) is located and removed automatically. The holomorphic 1-form, a pair of orthogonal vector fields, is then computed on the 3D colon surface mesh using the conjugate gradient method (see Fig. 18). The

colon surface is cut along a vertical trajectory traced using the holomorphic 1-form. Consequently, the 3D colon surface is conformal mapped to a 2D rectangle, and rendered using a direct volume rendering method accelerated with the GPU (see Fig. 19). The shape of the polyps is well preserved on the flattened colon images, which provides an efficient way to enhance the navigation of a virtual colonoscopy system.

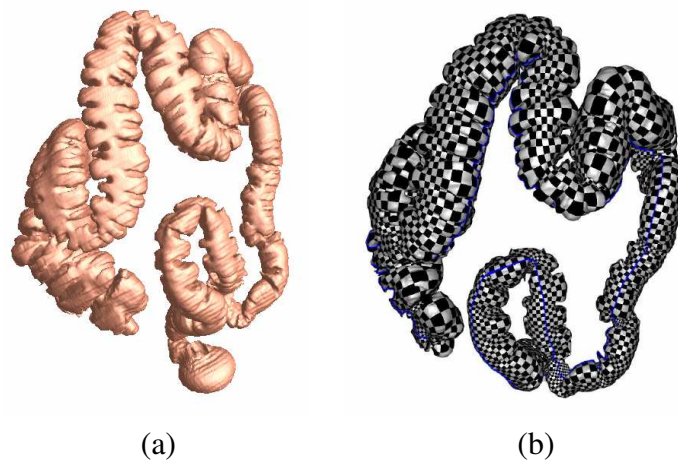


Figure 18: Conformal Virtual Colon Flattening: (a) The colon wall (b) The holomorphic one-form on the colon surface is illustrated using texture-mapping with a checker board image.

4.4.2 Metric-Driven N-RoSy Fields Design

4.4.2.1 Overview

Many objects in computer graphics and digital geometry processing can be described by *rotational symmetries*, such as brush strokes and hatches in non-photorealistic rendering, regular patterns in texture synthesis, and principle curvature directions in surface parameterizations and remeshing. N-way rotational symmetry (N-RoSy) has been proposed to model these objects.

The most important requirement for an N-RoSy field design system is to allow the user to fully control the topology of the field, including the number, positions and indices of the singularities, and the turning numbers of the loops [42,44]. Automatic generation of N-RoSy fields with user prescribed topologies remains a major

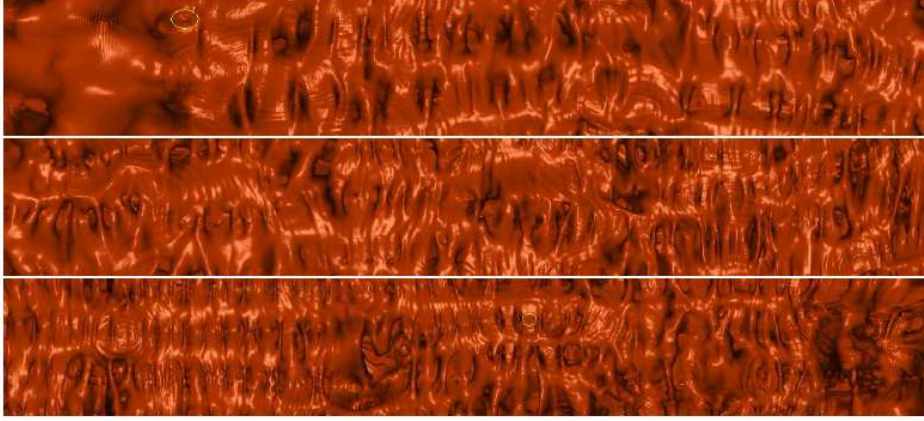


Figure 19: A flattened image for a whole colon data set is shown in three images, with polyps marked with small circles.

challenge. Fields with less singularities are often preferred, because singularities cause visual artifacts in real applications.

The method in [42] generates fields with user defined singularities, but it also produces excess singularities, which requires further singularity pair cancelation and singularity movement operations. The method in [44] is the first one that guarantees the correct topology of the field, but it requires the user to provide an initial field with all singularities at the desired positions. In practice, finding such an initial field is the most challenging step.

We provide a rigorous and practical method based on flat metric induced from surfaces' conformal structure which allows the user to design N-RoSy fields with full control of the topology and without inputting any initial field. Furthermore, the algorithm can automatically generate a smooth field with the desired topology and allow the user to further modify it interactively.

4.4.2.2 N-RoSy Fields Design

Our algorithms to design N-RoSy fields on general surfaces can be summarized as follows: 1. the user specifies the desired singularities of the vector field; 2. we compute surface conformal structure which induces a flat cone metric, such

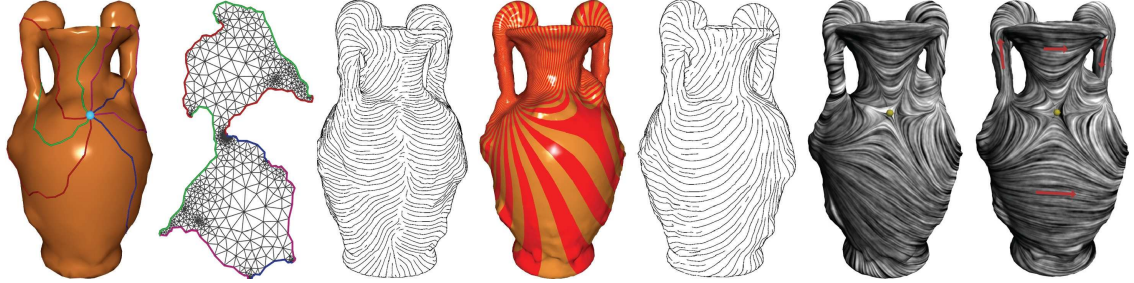


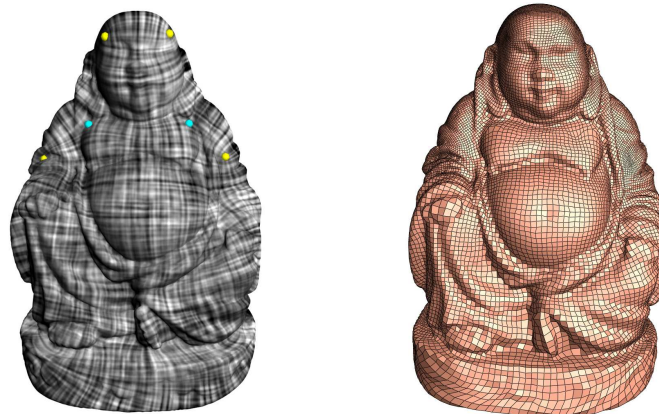
Figure 20: Algorithm pipeline (a). User specifies the desired singularities with both positions and indices. Here one singularity at the blue point with the index -2 . The curves are homotopy group basis. (b) We compute a flat metric, the curvature at the singularity is -4π , everywhere else 0. The surface is cut along the base curves and flatten to the plane. Note that the boundaries of the same color can match each other by a rigid motion. (c) Parallel vector field. The field has discontinuities along the red curve. (d) Compute a harmonic 1-form to compensate the holonomy. (e) The smooth vector field after rotation compensation. (f)(g) User inputs geometric constraints (red arrows) to guide the direction of the field, then the field is modified from (f) to (g).

that all the cone singularities coincide with those of the field; 3. we parallel transport a tangent vector at the base point to construct a parallel vector field; 4. if the parallel field has jumps when it goes around handles or circulates singularities, we apply two methods to eliminate the jumps: *rotation compensation* adjusts the rotation of the vector field; *metric compensation* modifies the rotation of the loops by deforming the surface. In the second stage, the vector field is further modified. we interactively edit the rotation and the magnitude of the vector field to incorporate user constraints. Figure 25 illustrates the pipeline using rotation compensation method.

4.4.2.3 Experimental Results

Remeshing We use the metric compensation method to adjust the metric to satisfy the tessellation compatibility condition. Then we develop the mesh to the plane, and tessellate the development. This induces a desired tessellation.

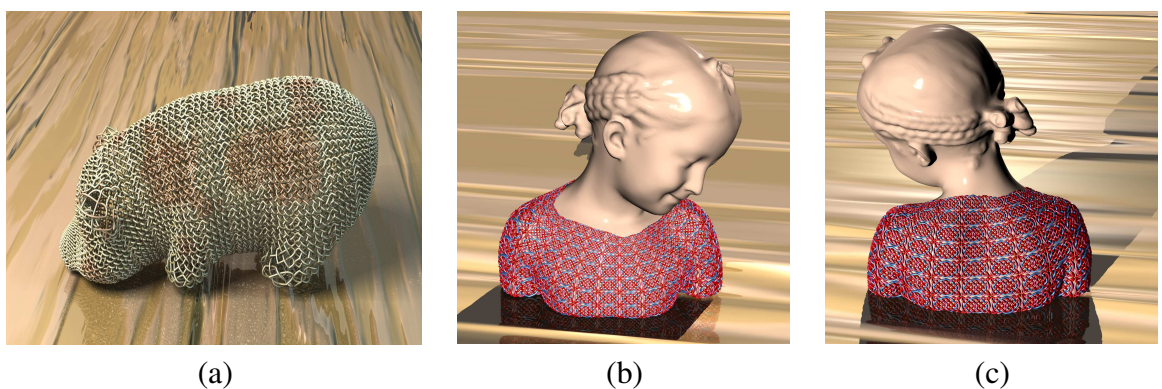
Figure 21 demonstrates the results of N-RoSy field on the buddha model. Frame (a) shows a 4-RoSy field on the buddha model, which induces a quad-remeshing as shown in (b).



(a) a 4-RoS field on buddha model (b) quad-remeshing

Figure 21

Celtic Knot on Surface Celtic knot refers to a variety of endless knots, which in most cases contain delicate symmetries and entangled structures. The local symmetry and the quality of remeshing of the surfaces play crucial roles for the knotwork on surfaces. Based on our remeshed results, those uniform quads and triangles provide a perfect canvas for Celtic knot design. Similar to the method in [32], we set control points directly on surfaces, connecting them using polynomials based on the knot designing rules. Compared with traditional geometric texture synthesis approaches, we do not need shell mapping from planar domains to surfaces. Figure 22 shows our Celtic knots synthesis results on several surfaces. The knotwork has complicated structures and rich symmetries.



(a)

(b)

(c)

Figure 22: Celtic knots designed surfaces

Pen-and-ink Sketching of Surfaces Pen-and-ink sketching of surfaces is a non-photorealistic style of shape visualization. Hatch directions can be treated as a 4-RoSy field. Since our method to construct 4-RoSy field enables the user to not only fully control singularities, but also edit the field interactively, our system is desirable for NPR applications.

For example, the editing process improves the hatching quality on the Bimba model shown in 23.

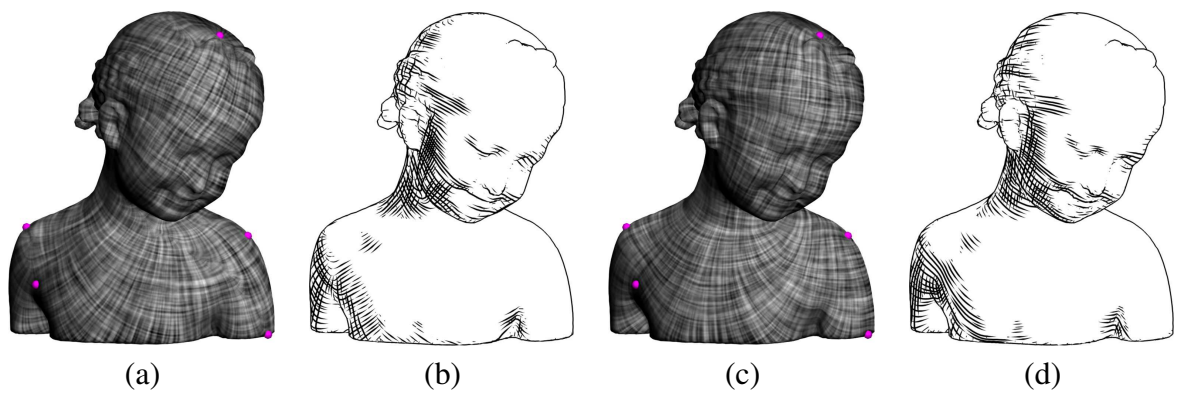


Figure 23: Pen-and-ink sketching of bimba before ((a) and (b)) and after editing ((c) and (d)). The hatch directions follow the natural directions better (e.g. neck,arm).

Chapter 5

Affine Structure

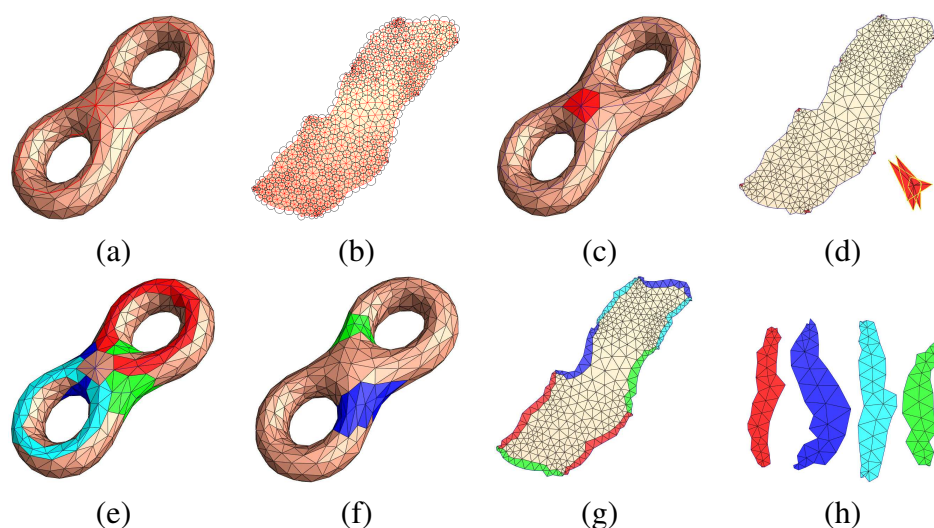


Figure 24: Affine atlas automatically acquired by using discrete Euclidean Ricci Flow. First, the user selects one singular point as shown in (a). Then a cut graph is labeled either manually or automatically as the dark curves in (a), where the cut graph is a set of canonical homology basis curves passing through the singular vertex. Second, the flat circle packing metric is computed using Ricci flow, illustrated in (b). The flat metric induces an planar embedding. The entire surface is sliced open along the cut graph to form a topological disk (i.e. fundamental domain). The interior of the fundamental domain is bijectively mapped to the plane. The mapping of the one-ring neighbor of the singular vertex is not 1 to 1, but $2g - 1$ to 1 shown in (d). Other charts covering the cut graph are constructed as shown in (e) and (f), and their overlapping relation with the central chart is shown in (g) and (h) by encoded colors, respectively.

5.1 Overview

For affine structure, X is R^2 , and G is the group of affine transformations. A 2 dimensional manifold M with an atlas $(U_\alpha, \phi_{\alpha})$, if all chart transition functions

$$\phi_{\alpha\beta} := \phi_\beta \circ \phi_\alpha^{-1} : \phi_{\alpha}(U_\alpha \cap U_\beta) \rightarrow \phi_\beta(U_\alpha \cap U_\beta)$$

are affine, then the atlas is called an affine atlas, and M is called an affine manifold.

Two affine atlases are compatible if their union is still an affine atlas. All the compatible affine atlases form an affine structure of the manifold. For closed surfaces, only genus-one surfaces have affine structures ([4,30]), but all open surfaces have affine structures.

One direct application for affine structure is manifold spline introduced in [15] based on polar form. They demonstrated the equivalence between manifold splines and the affine structure, and gave a systematic way to generalize splines defined on planar domains to manifold domains. They also gave the method constructing affine structures from conformal structures of surfaces with general topologies using discrete one-form method.

In this chapter, we first introduce another more flexible method to compute affine structures for surfaces with general topologies using discrete Euclidean Ricci flow (Section 5.2) [24]. Then we introduce our method to construct manifold spline for high genus surfaces with single singularity (Section 5.3.1) [14] instead of $2g - 2$ singularities in [15]. We present another method to construct manifold T-spline based on polycube maps which naturally induce surface affine structure (Section 5.3.2) [52].

5.2 Computing Surfaces Affine Structures

Surfaces affine structures can be induced from their conformal structures by quotient those singularity points, which has been proved in [15]. The method of constructing affine structures from conformal structures in [15] is using discrete one-form, which has to remove $2g - 2$ singularity points for genus g surfaces. Also the positions of singularity points are hard to control. Here, we introduce constructing affine structures from conformal structures using discrete Euclidean Ricci flow

method, which can reduce the number of singularity points to the theoretical minimum: one singularity point for any closed genus $g > 1$ surfaces and singularity point free for any open surfaces. This method is also flexible for position of the singularity point.

Given a surfaces Σ with arbitrary topology, we first compute its flat uniformization metric (Section 4.3.1) by concentrating all its curvatures to one vertex for closed high genus surfaces, or pushing curvatures to boundary vertices for open high genus surfaces, with all other vertices set to flat. Then we construct a family of open sets $\{U_\alpha\}$, such that the union of the open sets covers the surface Σ , $\Sigma \subset \bigcup U_\alpha$. Then we embed each U_α in the plane, as $\pi^{-1}(U_\alpha)$. The embedding gives each U_α of the local coordinates, namely ϕ_α . If one point $p \in \Sigma$ on the surface Σ is covered by two charts (U_α, ϕ_α) and (U_β, ϕ_β) , suppose $p_\alpha \in \bar{U}_\alpha$ and $p_\beta \in \bar{U}_\beta$, the chart transition map $\phi_{\alpha\beta}$ with the form

$$\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha^{-1},$$

where $\phi_{\alpha\beta}$ is an affine transformation. Therefore we construct an affine atlas $\{(U_\alpha, \phi_\alpha)\}$ which induces affine structure of the surface.

Figure 24 shows the flat uniformization metric of a genus two surface and its affine atlas.

5.3 Applications

5.3.1 Manifold Spline With Single Singularity

5.3.1.1 Overview

We develop a novel computational technique to define and construct powerful manifold splines with only one singular point by employing discrete Euclidean Ricci flow. The central idea and new computational paradigm of manifold splines are to systematically extend the algorithmic pipeline of spline surface construction from any planar domain to arbitrary topology. As a result, manifold splines can unify planar spline representations as their special cases. Despite their earlier success, the existing manifold spline framework is plagued by the topology-dependent, large number of singular points (i.e., $|2g - 2|$ for any genus- g surface), where the

analysis of surface behaviors such as continuity remains extremely difficult. The unique theoretical contribution of our method is that we devise new mathematical tools so that manifold splines can now be constructed with only one singular point, reaching their theoretic lower bound of singularity for real-world applications. Our new algorithm is founded upon the concept of discrete Ricci flow and associated techniques. First, Ricci flow is employed to compute a special metric of any manifold domain (serving as a parametric domain for manifold splines), such that the metric becomes flat everywhere except at one point. Then, the metric naturally induces an affine atlas covering the entire manifold except this singular point. Finally, manifold splines are defined over this affine atlas. The Ricci flow method is theoretically sound, and practically simple and efficient. We conduct various shape experiments and our new theoretical and algorithmic results alleviate the modeling difficulty of manifold splines.

Manifold Spline *Definition and Concept of Manifold Spline* A manifold spline of degree k is a triple (M, C, F) , where M is the domain manifold with an atlas. F is a map representing the entire spline surface. The knots are defined on M directly. C is the control point set, each control point is associated with a set of knots, such that

1. On each chart of the atlas, the restriction of F and C is a spline surface patch.
2. The evaluation of F is independent of the choice of the charts.

Equivalence to Affine Atlas The central issue of constructing manifold splines is that the atlas must satisfy some special properties in order to meet all the requirements for the evaluation independence of chart selection. Because the existing planar spline schemes are parametric affine invariant, this requires that all the chart transition functions are affine.

Existence in order to define a manifold spline, an affine atlas of the domain manifold must be found first. General closed 2-manifolds do not have an affine atlas. On the other hand, all open surfaces admit an affine atlas. In order to define manifold splines, the domain manifold has to be modified to admit an atlas by removing a finite number of points. This offers a theoretical evidence to the existence of singular points due to the topological obstruction. A classical result from characteristic class theory claims that the only closed surface admitting affine atlas is of

genus one.

5.3.1.2 Construction of Manifold Spline with Single Singularity

Figure 25 gives the pipeline to construct manifold spline with single singularity. The first step is to select singularity point; at the second step, connectivity around that singularity vertex is modified; the third step is to construct affine structure of given surface with arbitrary topology; manifold spline can be build based on the last step. We give detailed explanation for step one and two. Step three has been introduced in Section 5.2. For step four, we show some experimental results.

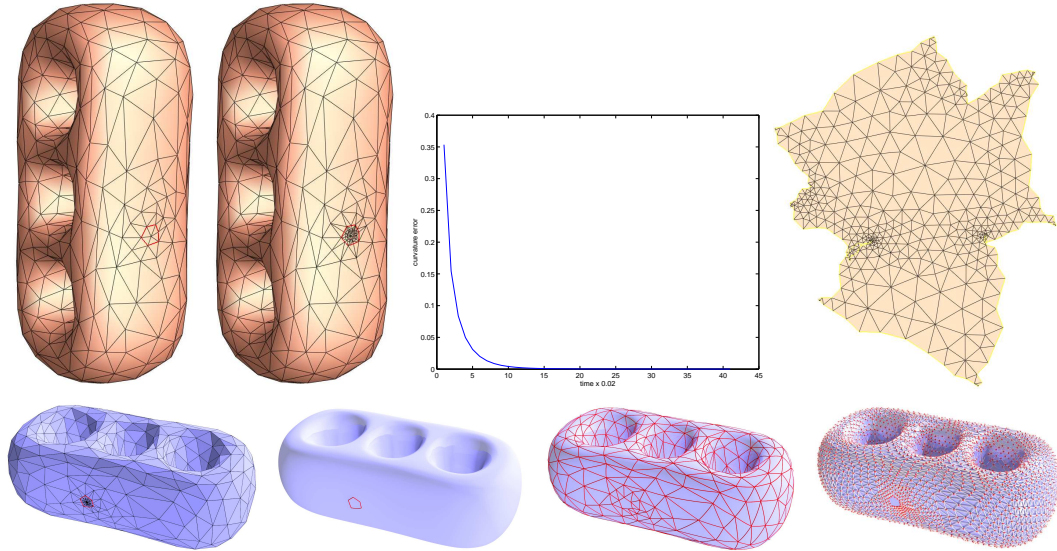


Figure 25: The pipeline to construct manifold spline with single singularity using discrete Euclidean Ricci flow. (a) Singularity Selection (b) Connectivity Modification (c) Ricci flow (d) One chart (e) Parametric domain (f) Manifold splines (g) Spline patchwork (h) Control point distribution.

Step 1: Selecting Singularities We can select the singular vertices $\{v_1, v_2, \dots, v_k\}, k \geq 0$ anywhere on the mesh arbitrarily, then we assign the target curvature of the singular vertices such that

$$\sum_{i=1}^k \bar{K}(v_i) = 2\pi\chi,$$

where χ is the Euler number the surface; the target curvature of other vertices are zero. Note that, there are several special cases that must be addressed.

- If the surface is a closed genus one mesh, then no singular vertex is selected.
- For a high genus mesh, we can select only one singular vertex and concentrate all curvature on it.
- If the mesh is open, we can assign the target curvatures for all the interior vertices to be zero and assign the target curvatures for boundary vertices such that the total boundary curvature equals to $2\pi\chi$. By this way, all the non-zero curvature will be pushed to the boundary.

Ricci flow only changes the vertex radii, therefore, the resulting metric is conformal to the original one, no angle distortion will be introduced. But the area distortion is unavoidable. The uniformity of the parameterization varies drastically depending on the choice of singularities. Our selection is based on the computation of area distortion for all possible cases, which sets every vertex as the single singularity in every case.

The area distortion error is measured with the following equation:

$$\frac{\sum_{i=1}^n (-\log(S_i) - \log(s_i))^2}{n}$$

where S_i is the i th triangle area in surface, s_i is its area in parametrization domain, and n is the total number of triangles. Figure 26(a) shows the distribution of the area distortion, blue area with low distortion, and red area with high distortion. In figure 26(b), the path of the arrow illustrates the rough behavior of the error distribution for the eight model.

When we sort the vertices with the area distortion errors, we may find the best position to put the singularity. Figure 27 gives five different cases of setting singularity. From the left to the right, the area distortion is increasing, with the left most being the best parameterized and the right most being the worst parameterized.

Step 2: Modify Local Connectivity around the Singular Vertex If both the initial curvature configuration and the target curvature configuration satisfy the constraints, any intermediate curvature configuration during Ricci flow will satisfy the

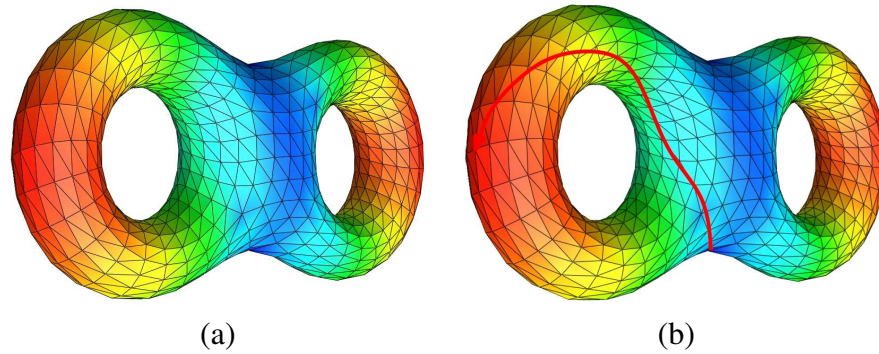


Figure 26: (a) The distribution of the area distortion is color encoded. (b) The path of the arrow illustrates the rough behavior of the error distribution for the eight model.

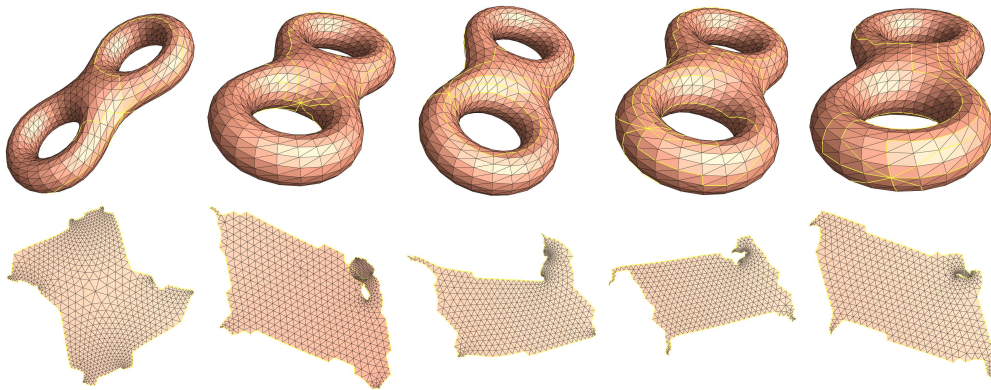


Figure 27: The first row gives the different positions of singularities on the same model; The second row shows their corresponding flat metric. From left to right, the area distortion increases.

constraints. Thus, it is enough to only consider the target curvature. If some singularities have high target curvature concentration, we need to modify the local connectivity in their neighborhoods.

We make the sampling in the neighborhood of the singularity much denser and increase the connectivity of the singularity no less than four times of genus, and vertices in the neighborhood have valence of about 6.

Step 4: Manifold Spline with Single Singularity We choose manifold triangular B -spline because of its flexibility in domain construction. This method can be also

applied to other manifold splines, such as T-splines and Powell-Sabin splines. The implementation details are described in [15].

One example of manifold triangular B -splines with single extraordinary is shown in Figure 28.

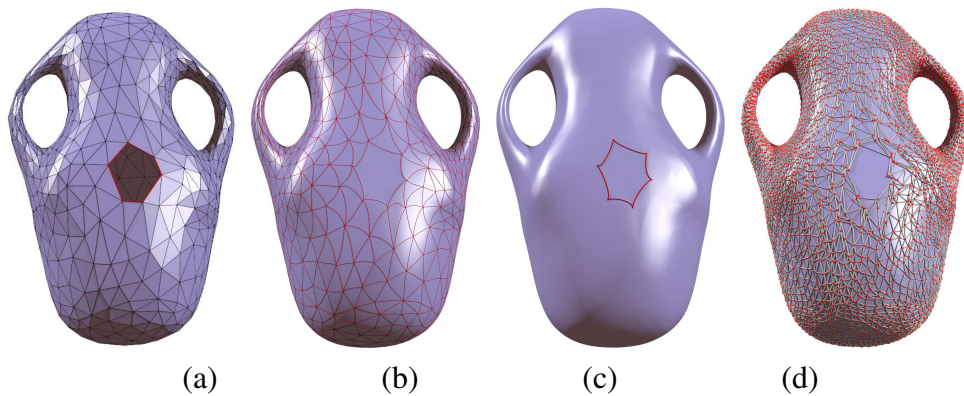


Figure 28: Example of manifold splines with single extraordinary point for closed high genus surfaces. (a) domain manifold with extraordinary point removed (b) manifold spline with extraordinary point filled (c) manifold spline with patchwork (d) manifold spline with control net

5.3.2 User-controllable Polycube T-spline

5.3.2.1 Overview

Polycube T-spline has been formulated elegantly that can unify T-splines and manifold splines to define a new shape representation for surfaces of arbitrary topology by using polycube map which naturally induces the affine structure with a finite number of extraordinary points. The data fitting quality using polycube T-splines hinges on the construction of underlying polycube maps. However, existing methods for polycube map construction exhibit some disadvantages. For example, existing approaches for polycube map construction either require projection of points from a 3D surface to its polycube approximation, which is therefore very difficult to handle the cases when two shapes differ significantly; or compute the map by conformally deforming the surfaces and polycubes to the common canonical domain and then construct the map using function composition, which is challenging

to control the location of singularities and makes it hard for the data-fitting and hole-filling processes later on.

We propose a novel framework of user-controllable polycube maps using discrete Euclidean Ricci flow, which can overcome the disadvantages of the conventional methods and is much more efficient and accurate. The current approach allows users to directly select the corner points of the polycubes on the original 3D surfaces, then construct the polycube maps using discrete Euclidean Ricci flow. We develop algorithms for computing such polycube maps, and show that the resulting user-controllable polycube map serves as an ideal parametric domain for constructing spline surfaces and other applications. The location of singularities can be interactively placed where no important geometric features exist. Experimental results demonstrate that the proposed polycube maps introduce lower area distortion and preserve small angle distortion as well, and subsequently make the entire hole-filling process much easier to accomplish.

5.3.2.2 Construction of Polycube Maps

Corner Selection Given a mesh M with arbitrary topology, user can design the polycube P based on the shape of the surface by directly selecting corners of P on M . The choices of the corners reflect the symmetry of M . The curvature at each corner c equals to $(2 - \frac{k}{2})\pi$, where k is the valence of c on the polycube p . Namely, protruding corners are with $\frac{\pi}{2}$, recessed corners are with $-\frac{\pi}{2}$. The total curvatures of all corners equals to $2\pi\chi(M)$, where $\chi(M)$ is the Euler-characteristic number of M . Figure 29(a) and (b) shows the selected corner points on Buddha model. The red corners are the protruding corners, the green corners are the recessed corners. For non-corner vertices, we set the curvature to be zeros.

Mesh Partition We use the discrete Euclidean Ricci flow to compute a new circle packing metric according to the target curvature. For any two corners c_1, c_2 on the mesh, whose correspondences are connected on the polycube, we compute the shortest path connecting them on the mesh under the new metric using Dijkstra's method (figure 29(c) and (d)). All such shortest paths segments partition the mesh to patches. Figure 30 (a) shows one patch from the partition of the buddha mesh by this step, which corresponds to one face of the polycube.

Rectification Each patch is a planar quadrilateral under the new metric, but

may not be a rectangle. We can use the Ricci flow method to rectify the planar quadrilateral to the rectangle by setting the target curvatures of 4 corners to be $\frac{\pi}{2}$, and all the other interior and boundary vertex curvatures to be zeros. Ricci flow can find a flat metric, the layout of the mesh under the flat metric is a rectangle. The aspect ratio of the rectangle is solely determined by original geometry of the patch. Figure 30 (b) illustrates the rectification result.

Polycube Assembly Assemble all the rectangles to a polycube, scale each rectangle along x and y directions when it is necessary. First, we build the dual graph of the polycube, each node represents a face of the polycube, each edge corresponds to an edge. Then we use breadth first searching method to traverse the dual graph. We first embed the root face, each time we access a new face, we determine the coordinates of its corners. In this way, we can embed the whole polycube in \mathbb{R}^3 . Figure 30(c) and (d) shows the polycube map for genus-0 buddha model.

If two rectangles on the polycube share one edge, make the corresponding vertices to align each other. Then we use a discrete harmonic map to relax the positions of the interior vertices of each rectangle with the fixed boundary condition.

In the above construction, the mapping between the polycube and the surface is automatically established. The shape of the polycube and the correspondence are fully determined by corner points. Therefore, the choices of the corner points are crucial. The followings are the important criteria for choosing the positions of the corners: the corners should be at regions with less geometric features for the purpose of better hole filling; the configuration of the corners should reflect the symmetry of the original surface.

5.3.2.3 Experimental Results: Polycube T-spline

Part of our experimental results in Figures 31 and 32 show that our method gives users more freedom to design the polycube; it induces lower area distortion between the surface and the polycube; it is capable to handle surfaces with more complicated topologies, such as high genus surfaces or open surfaces, which are difficult to handle by conventional methods.

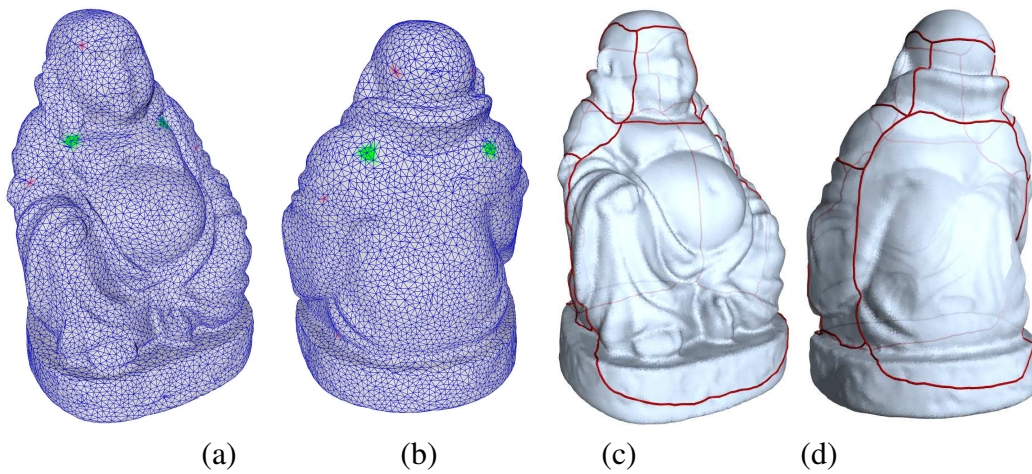


Figure 29: (a) and (b): Corner points are marked on Buddha model, red ones with $\pi/2$ target Gaussian curvature, and green ones with $-\pi/2$ target Gaussian curvatures. (c) and (d): Geodesics between corner points are marked with sharp edges, which are computed using Dijkstra's algorithm with computed conformal metric as edge lengths.

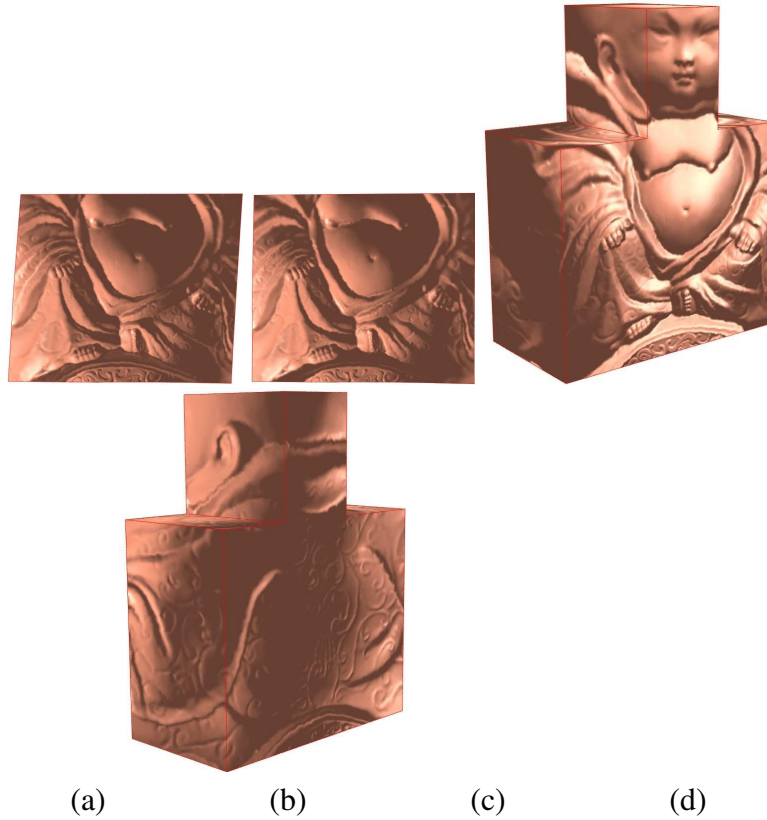


Figure 30: (a) and (b): One patch from Buddha model after partition, which corresponds to one face of the polycube. The one on the left is the quadrilateral before rectification, the one on the right is the rectangle after rectification. (c) and (d): User-controllable Polycube Map for Buddha Model.

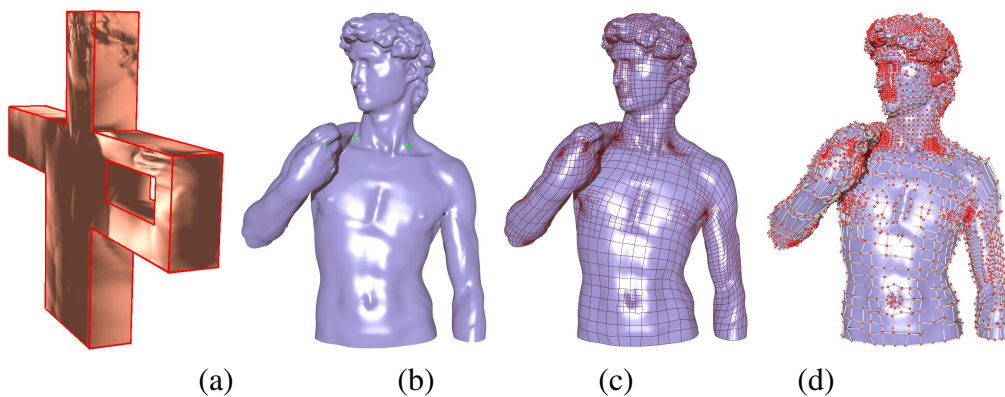


Figure 31: (a) User-controllable Polycube map (b) Polycube T-spline (c) T-junctions on polycube spline (d) Close-up of control points

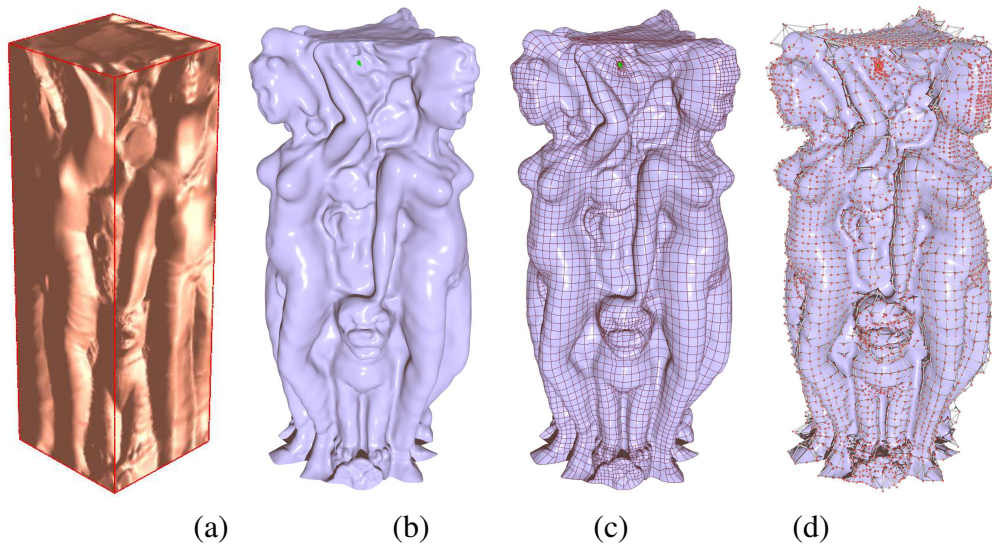


Figure 32: (a) User-controllable Polycube map (b) Polycube T-spline (c) T-junctions on polycube spline (d) Close-up of control points

Chapter 6

Hyperbolic Structure and Real Projective Structure

6.1 Overview

For hyperbolic structure, X is H^2 , and G is the group of Möbius transformations. For projective structure, X is RP^2 , and G is projective transformations. Hyperbolic structure was applied in [10] for topological design of surfaces, where the high genus surfaces were represented as quotient spaces of the Poincaré disk over Fuchsian group actions. In [13], Grimm and Hughes defined parameterizations for high genus surfaces and constructed functions on them. Wallner and Pottmann introduced the concept of spline orbifold in [51], which defined splines on three canonical parameter domains, the sphere, the plane and the Poincaré. But their works only consider the topology and ignore the geometry of the surface. For real applications, such as texture mapping, shape analysis and spline constructions, conformality between the original and the final metrics is highly desirable.

In this chapter, we first give the method computing surfaces hyperbolic structure and real projective structure using discrete hyperbolic Ricci flow (Section 6.2) [25]. Then we introduce shape space 6.3 [26, 28], using surfaces hyperbolic structure, which can be used as shape index for surfaces classification purpose.

6.2 Computing Surfaces hyperbolic Structure and Real Projective Structure

High genus surfaces have hyperbolic structures and real projective structures. Both of them can be induced from hyperbolic uniformization metrics on surfaces. The following algorithms are designed to compute the hyperbolic uniformization metric, hyperbolic structure and real projective structure for a given surface Σ with genus g greater than one.

1. Compute a canonical homology basis and canonical fundamental domain of the surface Σ .
2. Compute hyperbolic uniformization metric of the surface Σ using Discrete hyperbolic Ricci flow method.
3. Compute its Fuchsian group generators in the Poincaré disk model.
4. Construct a hyperbolic atlas.
5. Convert the hyperbolic atlas to the real projective atlas.

The algorithm in the first step has been introduced in [6].

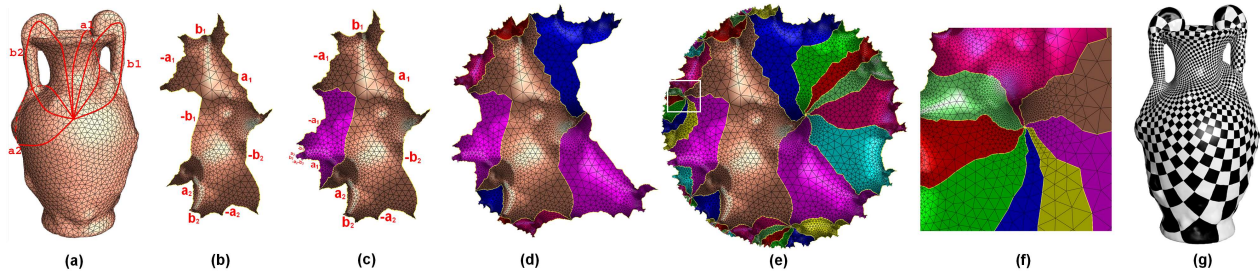


Figure 33: (a) Genus two vase model marked with a set of canonical fundamental group generators which cut surface into a topological disk with eight sides: $a_1, b_1, a_1^{-1}, b_1^{-1}, a_2, b_2, a_2^{-1}, b_2^{-1}$. (b) One period is conformally Embedded onto Poincaré disk with uniformization metric, marked with eight sides. (c) One Möbius transformation moves the side b_1 of one copy of the period matching the side b_1^{-1} of itself. (d) Eight Möbius transformations move sides of eight copies of the period matching the corresponding sides of itself. (e) A portion of tessellate of the vase in Poincaré disk. (f) Zoom in of the tessellation, showing no folding or overlapping. (g) Considering an Euclidean unit disk which is coincide with the Poincaré disk, the Euclidean coordinate of each vertex in (b) is used as texture coordinate for the vase model in (a), with check board texture.

6.2.1 Computing Hyperbolic Uniformization Metric

The *discrete hyperbolic Ricci flow* method is simple and powerful for computing the uniformization metrics of high genus surfaces. Compared with Euclidean Ricci flow, there are two major differences

1. Suppose a triangular face on the mesh with edge lengths. Instead of treating it as a triangle in the Euclidean space, we treat it as a triangle in hyperbolic space. Then all the angles in the triangle can be calculated using the hyperbolic cosine law:

$$\cosh l_{ij} = \cosh \gamma_i \cosh \gamma_j + \sinh \gamma_i \sinh \gamma_j \cos \phi_{ij}. \quad \mathbb{H}^2 \quad (31)$$

2. In the energy form in equation 17, let $u_i = \ln \tanh \frac{\gamma_i}{2}$, therefore the Hessian matrix of the energy f is

$$\frac{\partial^2 f}{\partial u_i \partial u_j} = \frac{\partial K_i}{\partial r_j} \sinh r_j.$$

The other parts of the algorithm are identical to those of the Euclidean Ricci flow. The hyperbolic Ricci energy is strictly convex, with a unique global minimum, which gives us the desired hyperbolic uniformization metric.

6.2.2 Computing Fuchsian Group Generators in the Poincaré Disk Model

This step aims to compute the canonical Fuchsian group generators used for computing the universal covering space and hyperbolic structure.

6.2.2.1 Computing Fundamental Group Generators

We first compute a set of canonical fundamental group generators $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$. Assume the base point is p , then a_i 's and b_j 's are closed loops through the base point. The surface S is sliced open along the fundamental group generators to form a topological disk F called the canonical fundamental domain. The boundary of F has the form $\partial F = a_1 b_1 a_1^{-1} b_1^{-1} a_2 b_2 a_2^{-1} b_2^{-1} \dots a_g b_g a_g^{-1} b_g^{-1}$.

6.2.2.2 Isometric Embedding in Hyperbolic Disk

We isometrically embed the universal covering space $\bar{\Sigma}$ onto the Poincaré disk using the uniformization metric computed from Section 6.2.1, and let $\phi : \bar{\Sigma} \rightarrow \mathbb{H}^2$ denote the isometric embedding.

We first select a face f_{012} from Σ arbitrarily. Suppose three edge lengths are $\{l_{01}, l_{12}, l_{20}\}$, and the corner angles are $\{\theta_0^{12}, \theta_1^{20}, \theta_2^{01}\}$ under the uniform hyperbolic metric. We simply embed the triangle as

$$\phi(v_0) = 0, \phi(v_1) = \frac{e^{l_{01}} - 1}{e^{l_{01}} + 1}, \phi(v_2) = \frac{e^{l_{02}} - 1}{e^{l_{02}} + 1} e^{i\theta_0^{12}}.$$

Then we can embed all the faces which share an edge with the first embedded face. Suppose a face f_{ijk} is adjacent to the first face, and vertices v_i, v_j have been embedded. A hyperbolic circle is denoted as (\mathbf{c}, r) , where \mathbf{c} is the center, and r is the radius. Then $\phi(v_k)$ should be one of the two intersection points of the two hyperbolic circles $(\phi(v_i), l_{ik})$ and $(\phi(v_j), l_{jk})$. Also, the orientation of $\phi(v_i), \phi(v_j), \phi(v_k)$ should be counter-clockwise. In the Poincaré model, a hyperbolic circle (\mathbf{c}, r) coincides with an Euclidean circle (\mathbf{C}, R) , satisfying

$$\mathbf{C} = \frac{2 - 2\mu^2}{1 - \mu^2|\mathbf{c}|^2} \mathbf{c}, R^2 = |\mathbf{C}|^2 - \frac{|\mathbf{c}|^2 - \mu^2}{1 - \mu^2|\mathbf{c}|^2},$$

where $\mu = \frac{e^r - 1}{e^r + 1}$. So the intersection points between two hyperbolic circles can be found by intersecting the two corresponding Euclidean circles. The orientation of triangles can also be determined using Euclidean geometry on the Poincaré disk.

We can continuously embed faces which share edges with embedded faces in the same manner, until we embed enough portion of the whole $\bar{\Sigma}$ onto the Poincaré disk.

6.2.2.3 Computing Fuchsian Group Generators

Given two pairs of points (p_0, q_0) and (p_1, q_1) in the Poincaré disk, such that the geodesic distance from p_0 to q_0 equals that from p_1 to q_1 . Then there exists a unique Möbius transformation ϕ , such that $p_1 = \phi(p_0)$ and $q_1 = \phi(q_0)$. ϕ can be constructed in the following way: construct a Möbius transformation ϕ_0 mapping

p_0 to the origin and q_0 to a positive real number, with

$$\phi_0 = e^{-i\theta_0} \frac{z - p_0}{1 - \bar{p}_0 z}, \theta_0 = \arg \frac{q_0 - p_0}{1 - \bar{p}_0 q_0}.$$

Similarly, we can define another Möbius transformation ϕ_1 , which maps p_1 to the origin, q_1 to a real number, and $\phi_1(q_1)$ equal to $\phi_0(q_0)$. Then the desired Möbius transformation ϕ is: $\phi = \phi_1^{-1} \circ \phi_0$.

Let $a_k, a_k^{-1} \subset \partial F$ are two boundary curve segments with their starting and ending vertices $\partial a_k = q_0 - p_0$ and $\partial a_k^{-1} = p_1 - q_1$, then the Möbius transformation $(p_0, q_0) \rightarrow (p_1, q_1)$ is the Fuchsian generator β_k corresponding to b_k . In fact, β_k maps a_k to a_k^{-1} . Similarly, we can compute α_k which maps b_k^{-1} to b_k . Therefore, we can compute a set of canonical Fuchsian group generators $\{\alpha_1, \beta_1, \alpha_2, \beta_2, \dots, \alpha_g, \beta_g\}$ corresponding to the set of canonical fundamental group generators $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$ computed from Section 6.2.2.1.

6.2.3 Constructing Hyperbolic Structure

With the computed universal covering space and the Fuchsian group generators, now we can construct the hyperbolic structure of the given surface now. First we construct a family of open sets $\{U_\alpha\}$, such that the union of the open sets covers the surface Σ , $\Sigma \subset \bigcup U_\alpha$. Then we locate a pre-image of each U_α in the universal covering space $\bar{\Sigma}$, as $\pi^{-1}(U_\alpha)$. The embedding of the pre-image $\pi^{-1}(U_\alpha)$ in the Poincaré disk gives the local coordinates of U_α , namely

$$\phi_\alpha := \phi \circ \pi^{-1}.$$

where ϕ is the embedding map for the universal covering space to the Poincaré disk. If one point $p \in \Sigma$ on the surface Σ is covered by two charts (U_α, ϕ_α) and (U_β, ϕ_β) , suppose $p_\alpha \in \bar{U}_\alpha$ and $p_\beta \in \bar{U}_\beta$, and a curve connecting p_α, p_β is denoted as γ . The homotopy class of $\pi(\gamma)$ is determined by p_α, p_β , denoted as $[\bar{p}_\alpha, \bar{p}_\beta]$. Assume $[\bar{p}_\alpha, \bar{p}_\beta] = \gamma_1 \gamma_2 \gamma_3 \dots \gamma_n$, where γ_k is one of the a_i 's or b_i 's, we replace a_i in $[\bar{p}_\alpha, \bar{p}_\beta]$ by α_i , b_i by β_i in γ to get the chart transition map $\phi_{\alpha\beta}$ with the form

$$\phi_{\alpha\beta} = \phi_1 \circ \phi_2 \circ \phi_3 \dots \phi_n,$$

where ϕ_j is one of the α_i 's or β_i 's. Therefore we construct a hyperbolic atlas $\{(U_\alpha, \phi_\alpha)\}$ which induces the hyperbolic structure of the surface.

6.2.4 Constructing Real Projective Structure

For a closed surface Σ with genus $g > 1$, its real projective atlas can be deduced from its hyperbolic structure (but the reverse is not true). Suppose $\{(U_\alpha, \phi_\alpha)\}$ is a hyperbolic atlas of Σ , then a real projective atlas $\{(U_\alpha, \tau_\alpha)\}$ can be straightforwardly constructed. Let

$$\tau_\alpha = \beta \circ \phi_\alpha \text{ and } \tau_{\alpha\beta} = \beta \circ \phi_{\alpha\beta} \circ \beta^{-1},$$

where β is the map from the Poincaré model to the Klein model defined in Eqn. 6.

Suppose ϕ_α has the form

$$\phi_\alpha = e^{i\theta} \frac{z - z_0}{1 - \bar{z}_0 z},$$

where $z_0 = x_0 + iy_0$, we use homogenous coordinates (xw, yw, w) to parameterize the points (x, y) on the Klein model, then the transition map $\tau_{\alpha\beta}$ has the following form: $\tau_{\alpha\beta} = \frac{1}{\lambda} OT$, where $\lambda = x_0^2 + y_0^2 - 1$, and O is the rotation matrix. O and T are:

$$O = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}, T = \begin{pmatrix} 1 + x_0^2 - y_0^2 & 2x_0y_0 & -2x_0 \\ 2x_0y_0 & 1 - x_0^2 + y_0^2 & -2y_0 \\ 2x_0 & 2y_0 & -1 - x_0^2 - y_0^2 \end{pmatrix} \neq 0, \quad (32)$$

6.3 Application: Shape Space

6.3.1 Overview

With the development of graphics hardware 3D scanning devices, the number of 3D geometric models in online repositories is dramatically increasing. The demand for effective retrieval of shape models is pressing. The primary challenge in building a shape-based classification and retrieval system is to find a computational representation of shape descriptors for which an index can be built, and similarity queries can be answered efficiently.

The problem of shape classification and comparison is very challenging. For a geometric algorithm, all the information that can be utilized is only the *topology*

and geometry of the shape. But for human beings, shape classification and comparison involves the expectations of the *functionalities* of the objects. For example, for a human observer, the slatted chairs can still be quite similar even if they have a different number of slats; but for a computer, the objects are quite different because they have different topologies. Low level algorithms based on the geometric information need to be developed first to lay down the foundation for high level methods, which are closer to the human intelligence. The algorithms in both levels have fundamental importance. This work focuses on the algorithms solely based on the geometric information.

Shape descriptors can be constructed using different levels of geometric information. For example, surfaces can be classified by their topological properties, such as the number of the handles and the boundaries. Shapes can be differentiated more precisely by differential geometric properties, such as principle curvatures and fundamental forms. Topological descriptors are global, succinct and intuitive, but less discriminating; whereas differential geometric descriptors are local, redundant, and counter intuitive, but much more discriminating. The huge storage requirements prevent differential geometric descriptors from practical applications. This work introduces a novel approach for shape classification and comparison, the descriptors are based on conformal geometry. Although they are not intuitive, conformal geometric descriptors are global, succinct and discriminating. In reality, it is hard to find two natural shapes with handles to share the same conformal descriptors. Conformal shape descriptors are intrinsic, independent of rotation, translation and scaling, also invariant with different triangulations and isometric deformation. They are stable, for deformations with small area stretching, like the posture change of a human skin surface, which changes slightly. They are efficient, easy to compute and compare. Therefore, we believe conformal geometric approach for shape classification and comparison has the potential for real applications.

6.3.2 Conformal Equivalence

A *conformal map*, also called an *angle-preserving map*, preserves local angles between two surfaces. We say two surfaces are *conformal equivalent* or with the same *conformal structure* if there exists a bijective conformal map between them.

Surfaces can be easily differentiated by the conformal equivalence. For example, the three human faces in Fig. 34 are topological equivalent, because all of them are two-holed annuli, but they are not conformal equivalent. This fact can be verified by checking their conformal descriptors. First, we conformally deform them onto the hyperbolic space as shown in the middle row in the figure, such that all boundaries are deformed to geodesics. Each surface is mapped to two congruent right-angled hyperbolic polygons. The edge lengths of the hyperbolic hexagon (determined by the lengths of 3 boundaries) form the conformal descriptor. The bottom row in figure 34 shows two genus one surfaces which are conformal inequivalent. The figure shows a map between them, which transfers the checker-board texture from the kitten surface to the rocker-arm surface. The right corner angles on the kitten surface are distorted on the rocker-arm, this shows the map is not conformal.

This work proposes to classify surfaces based on Teichmüller space theory. In this work, we only consider oriented surfaces. We use (g, r) to represent the topological type of the surface, g means the number of handles (genus), r the number of boundaries. Fix the topology of the surfaces, all conformal equivalent classes form a finite dimensional manifold, the so-called *Teichmüller* space [5]. The dimension of the Teichmüller space of (g, r) is $6g - 5 + 3r$, where each point represents conformal equivalence class, and the conformal map is homotopic to Identity. A curve connecting different points represents a deformation process from one class to the other. Figure 35 illustrates the concept. The teapot surface has one handle and one boundary at the spout, therefore is of topological type $(1, 1)$. The Teichmüller space is 3 dimensional. The teapot in the middle is twisted, the deformation process is indicated by the blue curve. The further away the curve goes, the greater the distortion is. Another deformation process is depicted by the red curve, the teapot is pulled taller and taller. The two deformation paths are illustrated in both \mathbb{R}^3 and the Teichmüller space.

We briefly summarize the Teichmüller spaces for surfaces with different Euler numbers. The Euler number of type (g, r) is $2 - 2g - r$. The computational algorithms for the Teichmüller coordinates of surfaces with non-negative Euler numbers have been introduced before. This work focuses on surfaces with negative Euler numbers.

- The Teichmüller space for $(0,0)$ type surfaces, namely genus zero closed

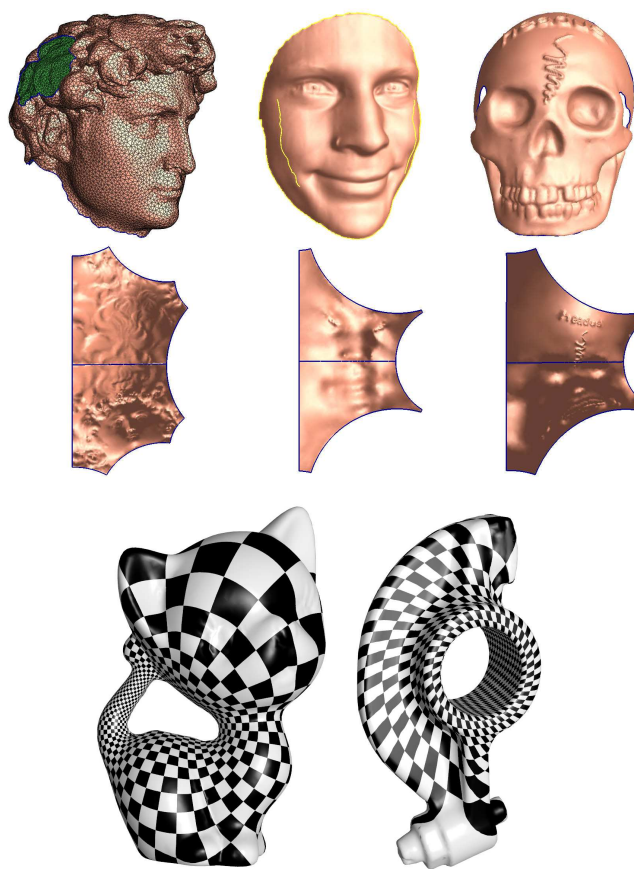


Figure 34: (a) Three human faces sharing the same topology (two holes annulus) are conformally mapped to hyperbolic space. (b) Map between genus one kitten model and torus.

surfaces, has only one point. That means, all genus zero closed surfaces are conformally equivalent. In this case, we conformally map the surface to the unit sphere. By mapping different surfaces to the unit sphere, we can easily construct the conformal mapping between two surfaces. The area distortion induced by the conformal mapping is called the *conformal factor*. In [16] we proved that the conformal factor and the mean curvature determine the surface unique up to a rigid motion in \mathbb{R} . We use area distortion and mean curvature as shape descriptors for shape comparison purpose in [16].

- The Teichmüller space for $(0, 1)$ type surfaces, namely genus zero surface with a single boundary, consists of a single point. All such surfaces can be mapped to the unit disk. Similarly, the conformal factor and mean curvature can be applied as shape descriptors, which determines the surface unique up to a rigid motion.
- The Teichmüller space for $(1, 0)$ type surface, namely tori, is two dimensional. The Teichmüller coordinates of a torus can be computed using global surface conformal parameterization method [19]. Basically, we can compute a holomorphic 1-form. By integrating the 1-form, we can map the universal covering space of the surface to the plane \mathbb{R}^2 . Each fundamental domain is mapped to a parallelogram. The Teichmüller coordinates of the torus are the length ratio between two adjacent edges of the parallelogram and the angle between two adjacent edges. We refer readers to [19] for details.
- For all the other surfaces, the Euler numbers are negative. The coordinates in Teichmüller space can be computed in the following method. First, there exists a unique Riemannian metric, called the *hyperbolic uniformization metric*, which is conformal to the original metric of the surface and induces -1 constant Gaussian curvature everywhere. Furthermore, all the boundaries become geodesics under the uniformization metric. Two closed curves are *homotopic*, if one can deform to the other without leaving the surface. Under the hyperbolic uniformization metric, each homotopy class has a unique geodesic. We choose a special set of homotopy classes on the surface, then compute the unique geodesic in each class, the lengths of these geodesics are Luo's coordinates [38], which form the length coordinates of the surface in Teichmüller space. This work focuses on the computation of the length

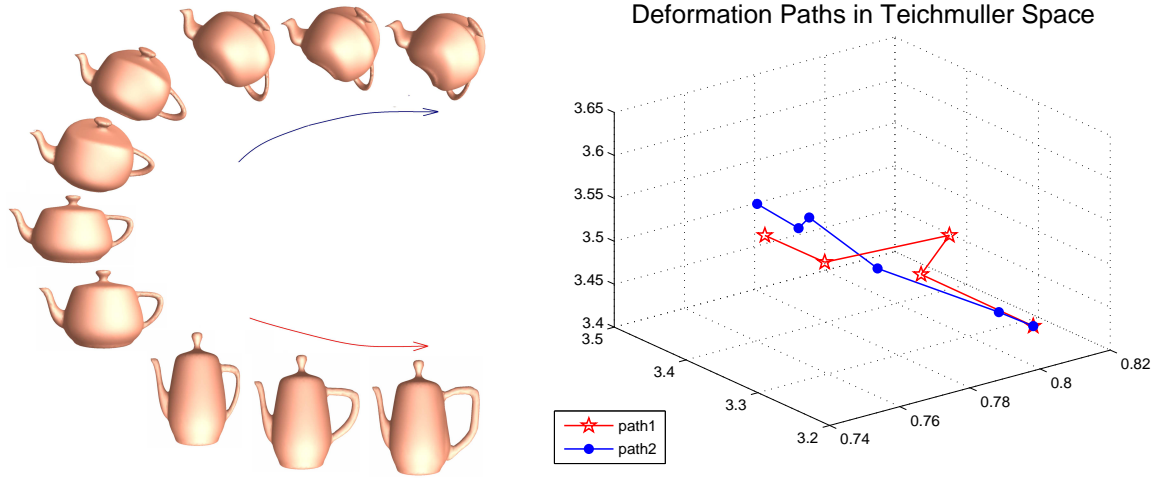


Figure 35: (a) Two groups of deformed teapot models. (b) Corresponding to two paths in Teichmüller space with each point representing a deformed model.

coordinates of surfaces with negative Euler numbers.

The major goal of this paper is to develop rigorous and practical algorithms to compute length coordinates of surfaces in Teichmüller space with negative Euler numbers. The major contributions of this work are

1. it proposes a theoretical framework to model all negative Euler number surfaces in a shape space, Teichmüller space. The framework has deep roots in modern geometry and is practical for computation. It offers novel views and tools for tackling engineering problems.
2. it introduces a series of practical algorithms for computing length coordinates of surfaces in Teichmüller space with complicated topologies. They classify surfaces according to their conformal class, which can be applied for shape indexing and classification.

We first introduce our algorithms to compute geodesic spectrum for general surfaces (Section 6.3.3) [26]; then generalize computing geodesic spectrum to computing *Teichmüller* coordinates (Section 6.3.4) [28].

6.3.3 Geodesic Spectrum

6.3.3.1 Overview

The geodesics are the locally shortest curves on surfaces. Computing geodesic paths on discrete setting has been intensively studied in the literature. The MMP algorithm [41] firstly provided an exact solution for the single source, all destination shortest path problem on a triangle mesh. Their algorithm partitions each mesh edge into a set of intervals (windows) over which the exact distance computation can be performed atomically. They proved a worst case running time of $O(n^2 \log n)$. [47] implemented this algorithm, and extend with a merging operation to obtain computationally efficient (running time $O(n \log n)$) and accurate approximations with bounded error. Other works include an exact geodesic algorithm with worst case time complexity of $O(n^2)$ described by [7] and partially implemented by [31], an algorithm for the single source, single destination geodesic path between two given mesh vertices, in $O(n \log 2n)$ time described by [33], and a variation of the fast-marching method to compute approximate geodesics on meshes in $O(n \log n)$ time by [35].

The geodesic lengths are closely related to metric, and reflect the global information of the surface. On general surfaces, there may be multiple geodesics in each homotopy class. For surfaces with uniformization metrics, the geodesics are unique in each homotopy class.

Theorem 8 (Geodesic Uniqueness) *Suppose (Σ, \mathbf{g}) is a closed compact surface with Riemannian metric \mathbf{g} , if Gauss curvature is neigative everywhere, then each homotopy class has a unique geodesic.*

The proof is based on Gauss-Bonnet theorem. We refer readers to [40] for details.

For our study of conformal structures, we can always deform the surface metric to the uniformization metric, then the geodesic lengths in each homotopy class form the length spectrum.

Definition 9 (Length Spectrum) *Let $(\Sigma, \bar{\mathbf{g}})$ be a surface with uniformization metric, the set of the lengths of closed geodesics on Σ is called the length spectrum of the surface Σ .*

The number of homotopy class of closed curves on a compact surface is countable. Since each homotopy class contains only one geodesic curve, also the length spectrum of a Riemann surface is countable. If two surfaces are conformal equivalent, they have same length spectra.

6.3.3.2 Computing Geodesic Spectrum of Genus Zero Surfaces

All closed genus zero surfaces are conformal equivalent. That means they are indistinguishable under their conformal structures. For genus zero surfaces with boundaries, their conformal structures are not identical any more. For example, for genus zero surfaces with three boundaries, there are infinite conformal equivalent classes, which form a 3 dimensional space.

Suppose Σ is a genus zero surface with three boundaries $\partial\Sigma = \{\gamma_0, \gamma_1, \gamma_2\}$. Under the hyperbolic uniformization metric \bar{g} , the three boundaries become geodesics, the conformal structure of Σ is determined by the geodesic lengths of the boundaries $\{l(\gamma_0), l(\gamma_1), l(\gamma_2)\}$.

There exist three geodesics $\{\eta_0, \eta_1, \eta_2\}$ perpendicular to the boundaries and separate the surface to two congruent hyperbolic hexagons with all right inner angles. The lengths of $\{\eta_0, \eta_1, \eta_2\}$ are determined by the lengths of $\{l(\gamma_0), l(\gamma_1), l(\gamma_2)\}$. Therefore the shape of the hexagon also determines the conformal structure of the surface.

Figure 36 shows three genus zero surfaces, each of them with three boundaries, and their congruent hyperbolic hexagons embedded in the Poincaré disk.

6.3.3.3 Computing Geodesic Spectrum of Genus One Surfaces

By the Uniformization theorem, a surface Σ with zero Euler number can be represented as $\mathbb{E}^2/Deck(\Sigma)$, where the deck transformation group consists of translations mapping the Euclidean plane \mathbb{R}^2 onto itself. The set of the lengths of closed geodesics on Σ can be computed algebraically from group $Deck(\Sigma)$. Therefore the whole procedure is to compute Euclidean Uniformization metric first, then the deck transformation group generators and finally the geodesic spectrum.

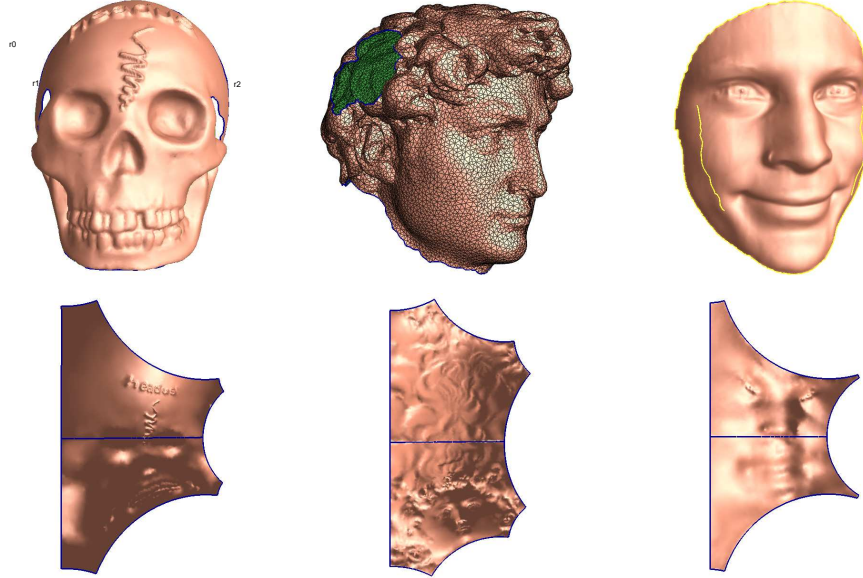


Figure 36: Conformal structure of the genus zero surface with 3 boundaries are determined by the lengths of the boundaries under the uniformization metric.

All elements in the fundamental group of Σ has the form $\gamma_{m,n} = ma + nb$, therefore the corresponding deck transformation is $\tau_{m,n} = \alpha^m \circ \beta^n$. All deck transformations are translations. We use a planar vector to represent each deck transformation. In practice, we normalize α to be $(1,0)$ by scaling and rotating on \mathbb{R}^2 , this won't affect the conformal structure of Σ , assume $\beta = (x,y)$, then $\tau_{m,n} = (m + nx, ny)$, the geodesic length in homotopy class $\gamma_{m,n}$ is $\sqrt{(m + nx)^2 + n^2y^2}$.

6.3.3.4 Computing Geodesic Spectrum of High Genus Surfaces

The computation for geodesic spectra of closed surfaces Σ with high genus number is very similar with those of genus one surfaces but more complicated.

Suppose a surface Σ is with negative Euler number, and \bar{g} is its uniformization metric. Then its universal covering space $\bar{\Sigma}$ with \bar{g} can be isometricly embedded in the hyperbolic space \mathbb{H}^2 . Each deck transformation $\tau : \mathbb{H}^2 \rightarrow \mathbb{H}^2$ is a Möbius transformation in upper half plane model of \mathbb{H}^2 . τ has a matrix representation, $\tau \in PGL(\mathbb{R}, 3)$,

$$\tau(z) = \frac{az + b}{cz + d}, a, b, cd \in \mathbb{R}, ad - bc = 1, z \in \mathbb{C}. \quad (33)$$

All deck transformations form the Fuchsian group of Σ .

Let γ be a closed curve and τ be the deck transformation corresponding to the homotopy class of γ , $\Phi(\tau) = [\gamma]$, then τ is a linear rational transformation as 33. The coefficient matrix of τ is

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

which has two real eigenvalues λ_1 and λ_2 . Let $\lambda = \max\{\lambda_1, \lambda_2\}$, then λ^2 is called the *multiplier of γ* . It can be proven that the unique geodesic homotopic γ is with length $2\ln\lambda$.

After computing the Fuchsian group generators, the geodesic spectrum of the high genus surface can be algebraically computed in a straightforward way. Suppose Σ is a closed high genus surface (if Σ is open, we use its double covering) with hyperbolic uniformization metric, $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$ is a set of fundamental group generators and $\{\alpha_1, \beta_1, \alpha_2, \beta_2, \dots, \alpha_g, \beta_g\}$ is the set of corresponding Fuchsian group generators. we want to compute the length of the geodesic in the homotopy class $\gamma = w_1 w_2 \dots w_n$, where w_k is one of the a_i 's or b_i 's. We replace a_i in γ by α_i , b_j by β_j in γ to get a Fuchsian transformation τ . For example, Suppose $\gamma = a_1 b_1 a_1^{-1} b_1^{-1}$, then its corresponding Fuchsian transformation is $\tau = \alpha_1 \beta_1 \alpha_1^{-1} \beta_1^{-1}$. We transform τ from Poincaré disk model to upper half plane model with $h^{-1} \circ \tau \circ h$.

Denoting the length of the geodesic homotopy to γ as l , the trace of the matrix τ satisfy the following relation: $tr(h^{-1} \circ \tau \circ h) = 2 \cosh(\frac{l}{2})$, then the geodesic length l is:

$$l = 2 \operatorname{acosh}\left(\frac{tr(h^{-1} \circ \tau \circ h)}{2}\right) \quad (34)$$

6.3.3.5 Results

The algorithms for computing geodesic spectrum from Fuchsian group generators are purely algebraic, easy to implement and fast to compute. We performed extensive experiments on general triangular meshes with complicated topologies.

The experimental results for genus one surfaces are shown in Figure 37, Figure 40(a) and table 2. Figure 37 shows six genus one closed surfaces. Table 2 lists

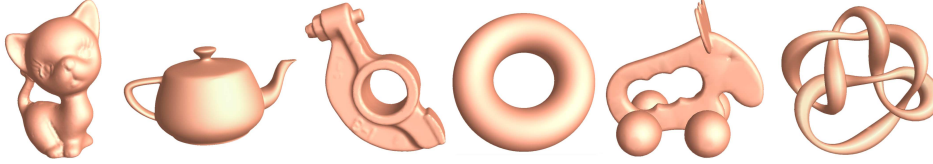


Figure 37: Genus One Models: Kitten model; Teapot model; Rocker Arm model; Torus model; Elk model; Knotty torus Model.

model	Geodesic Spectra of Genus One Surfaces							
Kitten	1.0000	2.0000	2.1716	2.3889	2.3926	2.9493	2.9552	2.9999
Teapot	1.0000	2.0000	3.0000	3.0275	3.1881	3.1886	3.6280	3.6289
Rocker Arm	1.0000	1.2951	1.6260	1.6463	2.0000	2.3687	2.3966	2.5902
Torus	1.0000	2.0000	2.2920	2.5007	2.5007	3.0000	3.0419	3.0419
Elk	1.0000	2.0000	3.0000	3.7649	3.8941	3.8968	4.2606	4.2657
Knot	1.0000	1.9999	2.9999	3.9999	4.9999	5.9999	6.9999	7.9999

Table 2: Geodesic Spectrum of Genus One Surfaces.

part of their geodesic spectra, lengths of the first 8 shortest geodesic. Figure 40(a) depicts the spectra for easier comparison purpose.

The experimental results for genus two surfaces are illustrated in Figure 39, Figure 41 and Table 4.

Genus three surface examples are demonstrated in Figure 38, Figure 40(b) and Table 3.

Because the geodesic spectrum indicates the conformal structures, the experimental results show that all the surfaces are conformally inequivalent.

The programs are coded in C++ and on Windows platform. The time consuming part of the pipeline is the computation of Ricci flow. For meshes with $30k$ faces, it takes around 2 minutes to compute the uniformization metric on a PC with $2GHz$ main frequency and $1G$ memory. The experimental results shows the algorithm is efficient and robust.

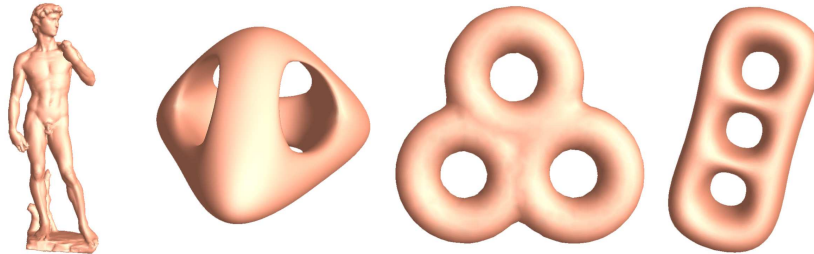


Figure 38: Genus Three Models: David model; Genus3 model; Three-Hole model; Holes3 model.

model	Geodesic Spectra of Genus Three Surfaces					
David	0.837531	1.031991	1.092184	1.675063	2.063981	2.184367
	3.369938	3.549802	3.728019	4.276527	4.402193	4.499322
Genus3	2.608251	2.750676	2.848831	3.841951	3.841954	3.841958
	3.841965	5.032769	5.216501	5.225431	5.290124	5.423762
Three-Hole	1.676075	1.690413	1.708316	3.319118	3.352150	3.380825
	3.393546	3.409811	3.416633	3.977163	4.040855	4.054275
Holes3	2.349592	2.354404	2.358875	2.376324	2.702315	3.588764
	3.605088	3.614748	3.675990	3.698398	4.699184	4.708807

Table 3: Geodesic Spectrum of Genus Three Surfaces.

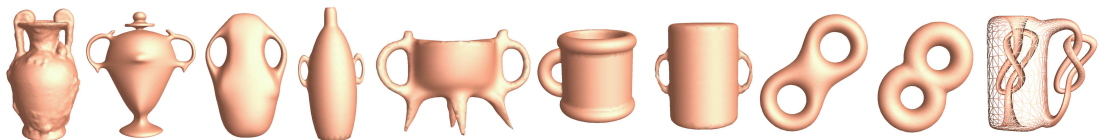


Figure 39: Genus Two Models: (a) Amphora model (b) World Cup model (c) Vase1 model (d) Vase2 model (e) Cup1 model (f) Cup2 model (g) Eight1 model (h) Eight2 model (i) Knotty model (j) Ding model

model	Geodesic Spectra of Genus Two Surfaces					
Amphora	1.490120	1.609724	2.656259	2.813498	2.980240	3.215555
	3.219448	3.367869	3.377852	3.396115	4.470360	4.829173
World Cup	0.664031	0.697533	1.328062	1.395066	1.992093	2.092599
	3.972944	4.071525	4.088756	4.104232	4.179075	4.187586
Vase1	0.985969	1.090920	1.971938	2.181839	2.957907	3.272759
	3.363453	3.570146	3.653931	3.675576	3.796436	3.835419
Vase2	1.128277	1.169550	2.256553	2.339099	2.890490	2.962423
	3.241003	3.251659	3.270609	3.315054	3.384830	3.508649
Ding	0.813765	0.830969	1.627529	1.661937	2.441294	2.492906
	3.629781	3.671329	3.785691	3.826148	3.827988	3.852406
Cup1	1.524061	1.843765	2.488011	2.894771	3.048121	3.320270
	3.336563	3.470016	3.476492	3.687530	3.976158	4.424202
Cup2	0.718974	0.762490	1.437948	1.524981	2.156923	2.287471
	3.820927	3.896118	3.931505	4.031120	4.038830	4.093943
Eight1	1.121486	1.142314	2.242971	2.284627	3.264045	3.301251
	3.364457	3.426941	3.569755	3.613231	3.619477	3.626651
Eight2	1.450139	1.461345	2.900279	2.922690	3.040154	3.059321
	3.512027	3.552606	3.587788	3.648212	4.350418	4.384035
Knotty	0.178828	0.180419	0.357656	0.360838	0.536484	0.541256
	2.932014	4.275073	7.142782	7.147029	7.155080	7.160804

Table 4: Geodesic Spectrum of Genus Two Surfaces.

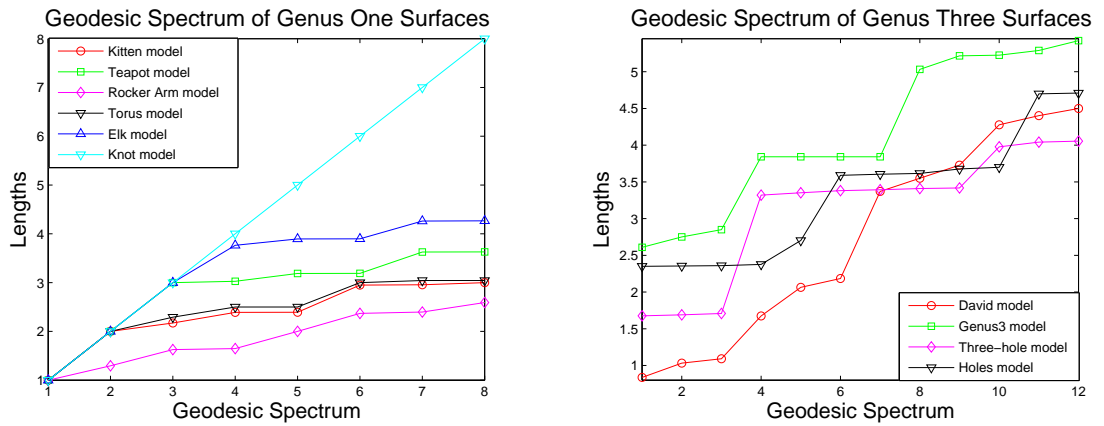


Figure 40: (a) Comparison of geodesic spectrum of genus one models. (b) Comparison of geodesic spectrum of genus three models.

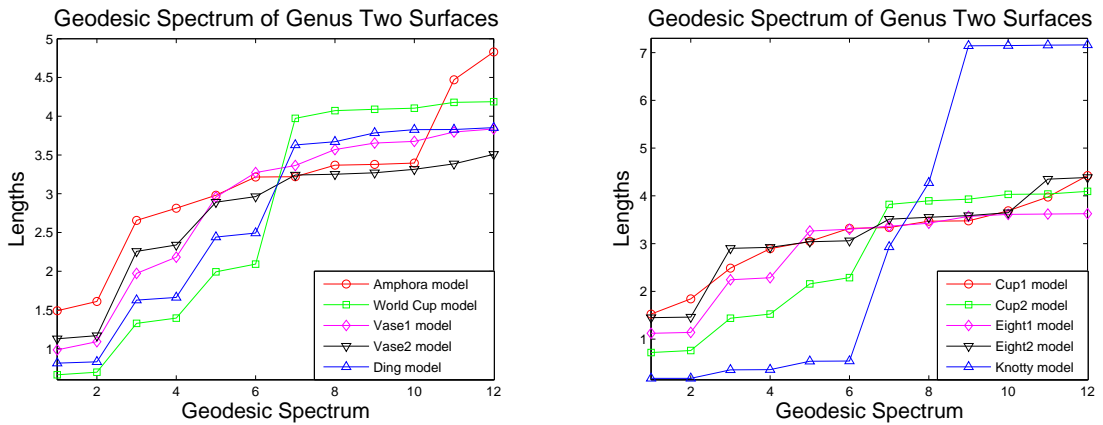


Figure 41: Comparison of geodesic spectrum of genus two models.

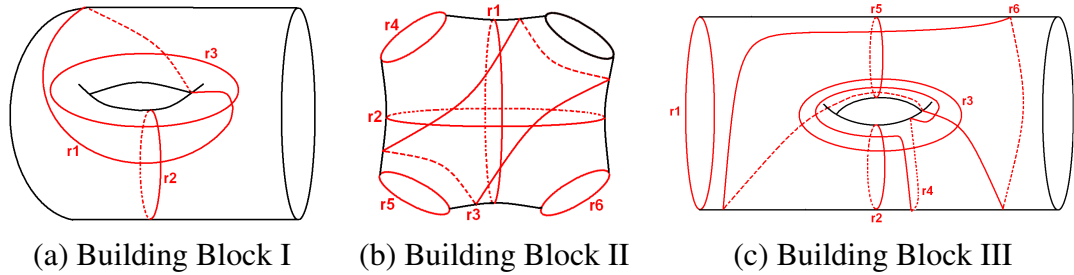


Figure 42: Basic Building Blocks. (a) Building block I: the geodesic lengths of red labeled curves determine its metric. (b) Building block II: the geodesic lengths of red labeled curves determine its metric. (c) Building block III: the geodesic lengths of red labeled curves determine its metric.

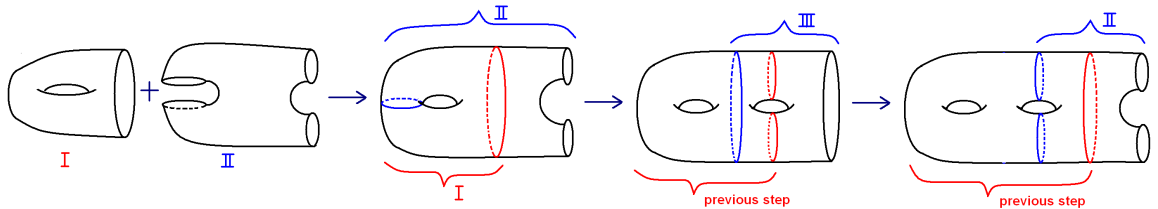


Figure 43: Using building blocks I, II and III to build all surfaces: using building block I and II to build genus one surface with two boundaries; continuously adding building block III to build genus two surfaces with one boundary; continuously adding building block II to build genus two surfaces with two boundaries. Repeating to get all surfaces.

6.3.4 Teichmüller shape space Coordinates

6.3.4.1 Overview

There are several coordinates defined in Teichmüller space. Here we adopt Luo’s coordinates in [38].

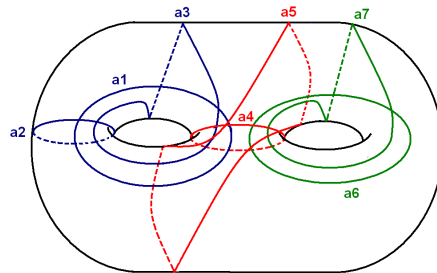


Figure 44: The geodesic lengths of the set of color labeled curves determine the metric of a genus two surface.

In the following discussion, we use $\Sigma_{g,r}$ to represent a surface Σ with topological type (g, r) , where g represents the genus, r means the number of boundaries.

Given a surface $\Sigma_{g,r}$ with negative Euler number, we can decompose the surface to three types of building blocks, as shown in Fig. 42. The procedure to build Σ from the building blocks is illustrated by Fig. 43. We use $I \cap II$ to denote the process to glue the block I to the block II . The gluing doesn't mean combining two blocks along their corresponding boundary curves, but by identifying their overlapping regions. For example, in the first gluing step in the figure, the overlapping region of I and II is a two-holed annulus. From left to right, we use basic building blocks I and II so that $I \cap II$ is homeomorphic to $\Sigma_{1,2}$, a genus one surface with two boundaries; continuously, adding building block III , so that $\Sigma_{1,2} \cap III$ is homeomorphic to $\Sigma_{2,1}$, a genus two surface with one boundary; then adding building block II , so that $\Sigma_{2,1} \cap II$ is homeomorphic to $\Sigma_{2,2}$, a genus two surface with two boundaries; repeat this procedure, we can generate all types of surfaces with negative Euler surfaces.

By this construction, a simple method is provided to define Luo's coordinates in Teichmüller space for general surfaces. For each building block, its conformal structure is determined by the lengths of geodesics homotopic to those red loops under the hyperbolic uniformization metric.

The geodesics are the locally shortest curves on surfaces. On general surfaces, there may be multiple geodesics in each homotopy class. For surfaces with hyperbolic uniformization metric, the geodesic is unique in each homotopy class, which can be proved by Gauss-Bonnet theorem.

When two building blocks are glued together to form a new surface, non-homotopic loops on the original blocks may become homotopic on the result surface. After canceling off the redundant loops, the lengths of geodesics homotopic to remaining loops determine the conformal structure of the result surface, which are the coordinates of this surface in Teichmüller space. For example, a closed genus 2 surface shown in Fig. 44, constructed from two building block I and one building block II , its Teichmüller coordinates are the lengths of geodesics homotopic to those loops marked with different colors. Loops with same color indicate that they come from same building block. In general, for a surface $\Sigma_{g,r}$ with a negative Euler number, their Teichmüller coordinates are determined by the lengths of $6g + 3r - 5$

closed geodesics.

6.3.4.2 Algorithms of Computing Teichmüller Shape Space Coordinates

Teichmüller coordinates are obtained by measuring the lengths of geodesics homotopic to a group of loops on surfaces under hyperbolic uniformization metric, and the geodesics are unique in each homotopy class since Gauss curvature is constant negative everywhere.

Algorithm 1 Compute Teichmüller Coordinates

1. Decompose the surface to building blocks.
 2. Determine the homotopy classes of the geodesics.
 3. Compute the lengths of the geodesics in each homotopy class.
-

In the above steps, the algorithms to compute the canonical homology basis $\{a_1, b_1, a_2, b_2, \dots, a_g, b_g\}$ and the corresponding Fuchsian group generators $\{\alpha_1, \beta_1, \alpha_2, \beta_2, \dots, \alpha_g, \beta_g\}$ have been introduced in Section 6.2.2. The goal is to compute the lengths of geodesics homotopic to those loops marked in Fig. 42. After redundant loops belonging to the same homotopic class removed out, for example, the remaining loops for a closed genus two surface is depicted in Fig. 44.

To compute the length of geodesic homotopic to a loop γ on surface, we first use the algorithm in [29] to determine its homotopy class, which can be symbolically represented, for example: $\gamma = a_1 b_1 a_1^{-1} b_1^{-1}$. Then by mapping each a_i to α_i and b_j to β_j , we get its representation using corresponding Fuchsian transformations, still the previous example: $\phi_\gamma = \alpha_1 \beta_1 \alpha_1^{-1} \beta_1^{-1}$. Let the length of γ denoted as l_γ , and we use the matrix representation of ϕ_γ on the upper half plane. l_γ can be easily computed from the following relation: $|tr(\phi_\gamma)| = 2 \cosh(\frac{l_\gamma}{2})$.

We have implemented the algorithms for computing the length coordinates in Teichmüller space using C++ on the Windows platform. We verify our method by computing the shape coordinates on a large number of surface models with various topologies. The resolutions of the models range from thousands to tens of thousands of triangular faces. Due to the page limit, we only list part of our experimental results.

Robustness Length coordinates in Teichmüller space are intrinsic properties of surfaces, independent of translation, rotation, scaling, and also insensitive to local noises, insensitive to the resolutions of the surface. We tested the robustness of our algorithm by computing one model with different resolutions. Figure 45 illustrates one such example. The vase model is triangulated using different resolutions, and the number of faces are $5k$, $10k$, $20k$ and $40k$ respectively. We tested our Teichmüller coordinates algorithm, the results are listed in the table 5. The relative error is less than 0.3%.

Surface Indexing and Classification Teichmüller coordinates can be directly applied for indexing and classification of surfaces with the same topology. The distance among shapes in the Teichmüller space can be approximated directly using the Euclidean distances among their Teichmüller coordinates. In our experiments, we tested genus two closed surfaces and genus three closed surfaces.

For closed genus two surfaces, the dimension of Teichmüller space is seven. The Teichmüller coordinates for eight genus two teapot models are illustrated in Fig. 46. The distances in the Teichmüller space among 23 genus two surfaces are listed in the Table 6. We cluster the shapes according to their Teichmüller distance. For example, table 7 shows a neighborhood of the shape of the teapot7 model in the Teichmüller space. The surface closest to the teapot7 looks very similar to it. This matches our intuition. Another example is shown in Table 8, which gives the neighborhood of the eight model in Teichmüller space. Again, the closest model to eight is the one that looks most similar to it. Furthermore, by examining the Table 6, we can also find that the knotty bottle model (the fifth model of the first row) is further away from all the others in the Teichmüller space, because its geometry is quite different from the others. Therefore, Teichmüller coordinates match our intuition.

Vase Model	Coordinates of Vase Model						
	1st	2nd	3rd	4th	5th	6th	7th
Face #: 5k	3.55027	0.99990	3.88055	5.55885	6.11438	3.33029	3.66071
Face #: 10k	3.55700	0.99832	3.88144	5.55611	6.11180	3.33369	3.66703
Face #: 20k	3.55805	0.99759	3.88316	5.55517	6.11112	3.33357	3.66713
Face #: 40k	3.55905	0.99559	3.88416	5.55417	6.11012	3.33367	3.66813

Table 5: Comparison of Coordinates of Vase Model with Different Densities. The dimension of Teichmüller space coordinates for closed genus two surfaces is seven.

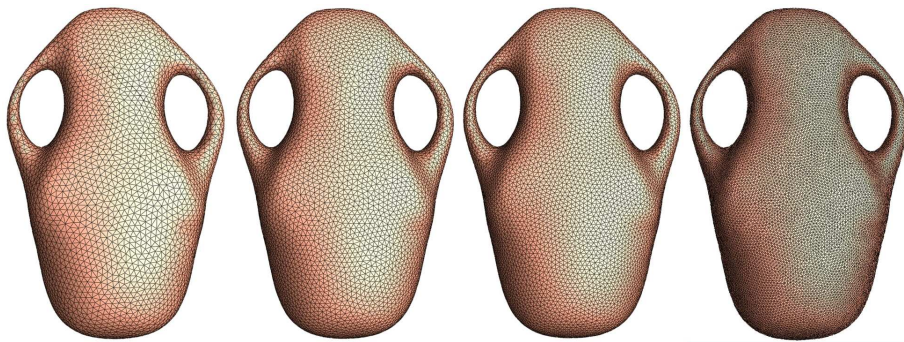


Figure 45: Same model with different triangulation density: 5k, 10k, 20k and 40k. Comparison of Teichmüller space coordinates with different densities is listed in table 5.

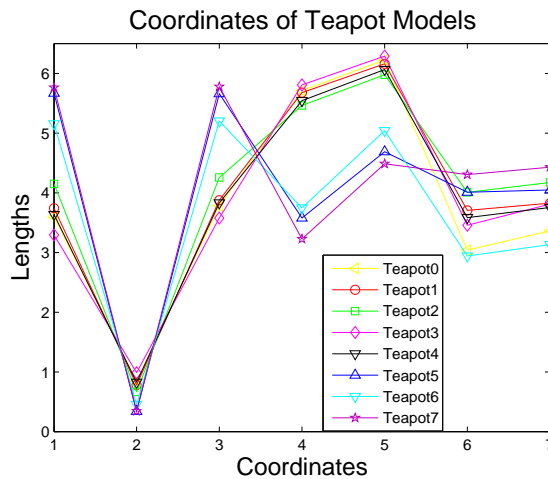


Figure 46: The dimension of Teichmüller space coordinates for closed genus two surfaces is seven. Here we visualize the Teichmüller space coordinates for teapots listed in table 7.















































Dist.																								
	4.09	5.39	3.22	4.49	6.69	2.22	2.16	2.21	2.39	2.32	4.38	3.87	4.88	3.51	3.52	3.42	3.25	4.02	3.29	3.20	3.44	3.26	3.41	
		7.75	3.06	6.74	9.98	2.11	2.37	2.82	2.05	2.14	4.59	3.68	4.97	2.76	2.84	3.01	3.59	2.81	2.99	3.64	4.78	3.73	4.30	
			4.95	1.04	7.92	6.62	5.95	5.62	6.17	6.17	7.27	7.89	7.34	7.46	6.56	6.48	5.61	6.97	6.32	5.63	4.72	5.56	5.23	
				3.94	8.99	2.47	1.73	1.64	1.85	1.83	4.27	4.27	4.56	3.08	3.43	3.22	2.37	3.84	3.02	2.52	2.75	2.43	2.64	
					7.79	5.61	4.92	4.59	5.16	5.14	6.35	6.91	6.45	5.44	5.56	5.48	4.57	6.00	5.30	4.60	3.71	4.53	4.21	
						8.48	8.48	8.35	8.61	8.62	9.27	9.13	9.57	8.75	8.65	8.68	8.48	8.95	8.57	8.40	8.08	8.43	8.31	
							0.83	1.51	0.73	0.71	4.05	3.15	4.51	2.09	1.92	1.51	2.27	2.42	1.51	2.36	3.59	2.43	3.12	
								0.79	0.65	0.25	3.72	3.15	4.17	1.79	2.30	1.86	1.44	2.88	1.48	1.60	2.83	1.63	2.37	
									1.40	0.92	3.10	2.83	3.52	2.51	3.02	2.64	1.54	3.62	2.14	1.26	2.13	1.28	1.61	
										0.59	4.28	3.60	4.53	2.36	2.76	2.40	2.95	2.28	2.51	2.18	3.38	2.21	2.96	
											3.71	3.07	4.16	1.63	2.20	1.76	1.62	2.76	1.30	1.74	3.01	1.79	2.52	
													1.62	0.64	5.21	5.71	5.40	4.61	6.26	4.83	4.30	4.03	4.35	3.88
														1.19	4.43	4.86	4.58	4.24	5.33	4.10	3.97	4.30	4.06	3.94
															5.67	6.19	5.88	5.02	6.73	5.31	4.92	4.82	4.86	4.63
																0.37	0.62	1.14	1.14	0.62	1.30	2.55	1.36	1.61
																	0.63	1.63	0.63	0.96	1.80	2.97	1.87	2.58
																		1.25	1.09	0.60	1.45	2.68	1.49	2.23
																			2.20	0.87	0.34	1.47	0.29	1.03
																				0.53	1.39	3.57	2.45	3.18
																					0.96	2.22	1.04	1.75
																						1.28	0.18	0.80
																							1.24	0.57
																								0.76

Table 6: Distances between genus two surfaces in Techmüller space.









Models	Teapot5	Teapot6	Teapot2	Teapot4	Teapot1	Teapot0	Teapot3
distance							
	0.6468	1.1923	3.5202	4.1694	4.1742	4.5179	4.53

Table 7: The sorted distances between teapot7 and other genus two models in Techmüller space. Here we only show the closest ones.










Distance								
	0.3732	0.6223	0.6239	1.1427	1.1439	1.3072	1.3664	1.6182

Table 8: The sorted distances between eight and other genus two models in Techmüller space. Here we only show the closest ones.

Chapter 7

Spherical Structure and Topological Structure

7.1 Spherical Structure

Closed genus zero surface has spherical structure, where X is S^2 and G is the group of rotation, which can be induced from its conformal map to the unit sphere.

One 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 [17]. The spherical uniformization metric and spherical structure can be induced by these conformal maps.

Another method is to directly compute spherical uniformization metric of closed genus zero surfaces using discrete spherical Ricci flow, then embed onto unit sphere, which induces spherical structure of surfaces [24]. Compared with discrete Euclidean Ricci flow and discrete hyperbolic Ricci flow, there are two major differences for discrete spherical Ricci flow:

1. Suppose a triangular face on the mesh with edge lengths. Instead of treating it as a triangle in the Euclidean space or hyperbolic space, we treat it as a triangle in unit sphere. Then all the angles in the triangle should be calculated using the spherical cosine law:

$$\cos l_{ij} = \cos \gamma_i \cos \gamma_j - \sin \gamma_i \sin \gamma_j \cos \phi_{ij}. \quad \mathbb{S}^2 \quad (35)$$

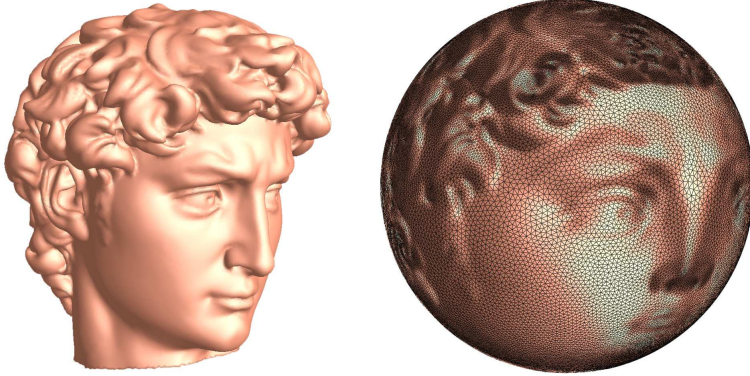


Figure 47: David head model mapped to unit sphere using computed spherical uniformization metric.

2. In the energy form in equation 17, let $u_i = \ln \tan \frac{\gamma_i}{2}$, therefore the Hessian matrix of the energy f is

$$\frac{\partial^2 f}{\partial u_i \partial u_j} = \frac{\partial K_i}{\partial r_j} \sin r_j.$$

Figure 47 shows David head model mapped to unit sphere using computed spherical uniformization metric.

7.2 Topological Structure

Arbitrary topology surface has topological structure, where X is R^2 , and G is the group of Homeomorphisms. Inspired by the rich theory of topology as well as the existing works on surface mapping, we present a novel solution to the problem of computing continuous maps with different homotopy types between two arbitrary triangle meshes with the same topology in [6] (Section 7.3).

7.3 Application: Topology-based Surface Mapping

Surface mapping is of prime significance in many graphics applications including shape analysis, texture mapping, animation transfer, shape morphing, feature registration, and many other digital geometry processing methods. Although

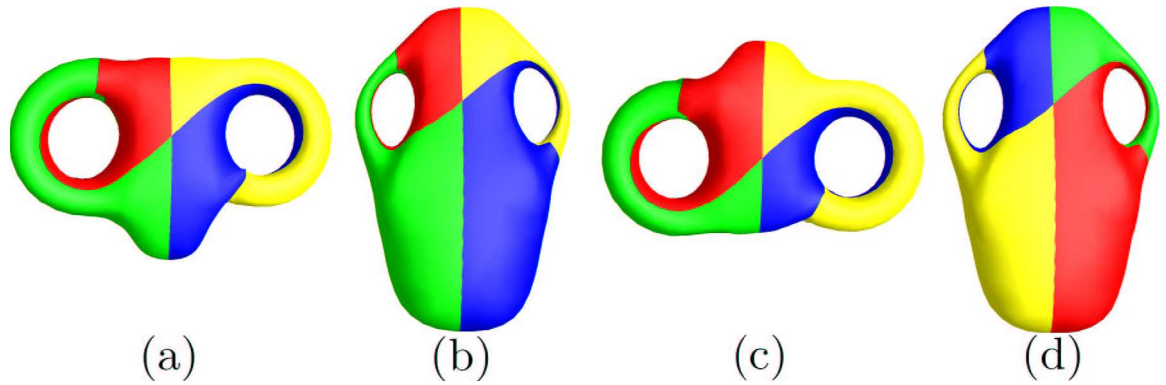


Figure 48: Visualization of two different homotopy classes of mappings between the vase and the two-hole torus by color coding. Each mapping is produced by a different homology basis on each surface. (a) and (b) give the mapping between the two-hole torus and the vase with handles right to right, left to left. While we obtain a different mapping between them in (c) and (d) with the handles right to left, left to right.

topological concepts and techniques have been broadly applied in computer graphics and geometric modeling, the homotopy type of a mapping between two surfaces has not been addressed before. While other surface mapping methods focus on a single homotopy class, we articulate a theoretically rigorous method that produces many continuous maps of different homotopy type between two arbitrary triangle meshes with the same topology (see Figure 48).

The intent of our method is to create a continuous, piecewise-linear map between triangulated surfaces with the same genus and number of boundaries. We first obtain a common domain for the surfaces, from which to extract the final mapping.

To obtain this, we first compute a special set of curves (edge paths), a canonical homology basis, for the surface (see Fig. 49). This set of curves defines a regular cut graph, along which we slice each surface open in a canonical way. The homotopy type of the mapping is governed by the choice of a homology basis for each surface, as well as the one-to-one matching between these sets of curves. This will produce a mapping that belongs to a unique homotopy class of surface mappings.

Once we have the two homology basis sets and a matching between them, the surface can be parameterized over the canonical fundamental domain M^* . Every surface with the same genus can be cut open to a disk in this manner. The overall alignment of mesh features is governed by the way in which the curves on two

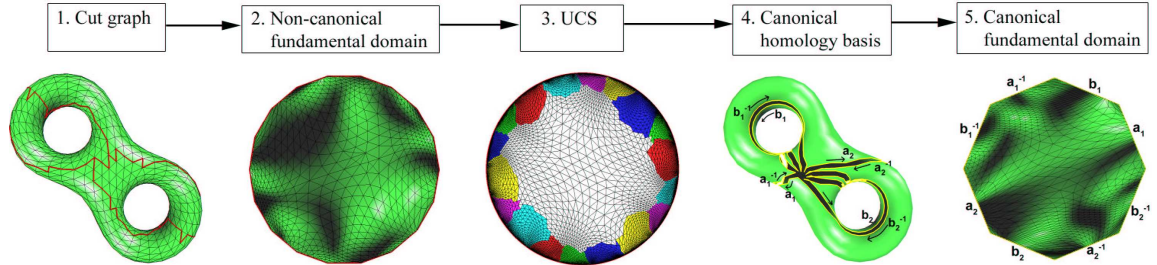


Figure 49: Procedure for computing a canonical fundamental domain, M^* . Step 1: Compute an irregular cut graph that will open the surface to a single disk. Step 2: Slice the surface open to a non-canonical fundamental domain, whose 18 edges are segments of homology basis curves. Step 3: Glue copies of the domain to itself along corresponding segments to form the UCS \tilde{M} . Choose a vertex $v \in M$ and a preimage of v , $\tilde{v}_0 \in \tilde{N}$. Trace paths between \tilde{v}_0 other preimages $\tilde{v}_k \in \tilde{M}$ of v , each of which corresponds to a single homology class of curves on M . Step 4: Find a combination of such curves that forms a canonical homology basis. Step 5: Slice the surface along this cut graph, constructing a canonical fundamental domain M^* . The i th handle is sliced open along the curve sequence $a_i, b_i, a_i^{-1}, b_i^{-1}$.

surfaces correspond. In our system, the curve mappings can be controlled by the user before and after the mapping to the plane. Once we have cut the surfaces open, we parameterize them onto the plane. Then, we align these planar domains and extract the mapping from the two sets of vertices and faces (see Fig. 50).

With a simple rotation that re-aligns the boundary edges of the polygonal domain regions (which translates to curve matching on the original surfaces), we can easily change the homotopy type, or global structure, of the mapping (see Fig. 51).

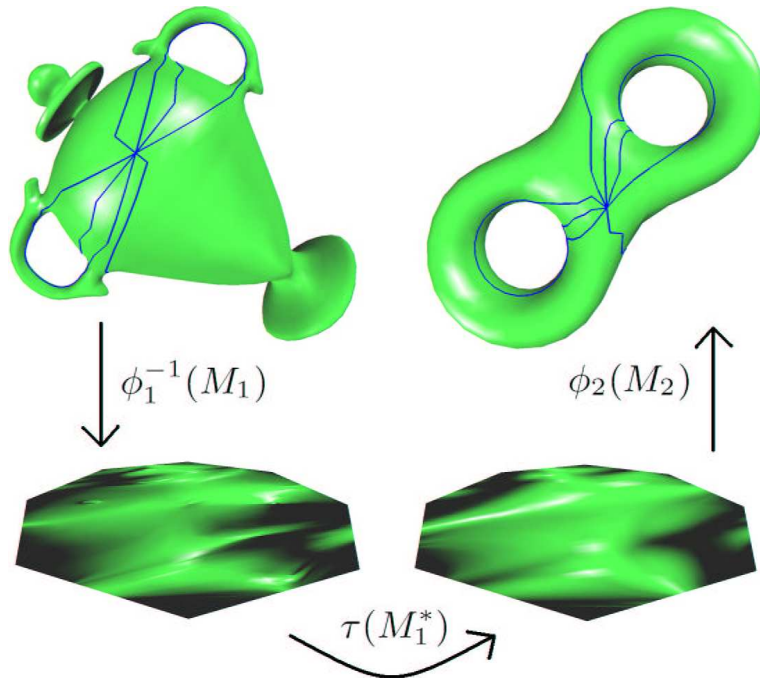


Figure 50: An overview of the surface mapping process. The vase and the two-hole torus are cut open along a common set of curves and parameterized over the canonical fundamental domain M^* . The final mapping is extracted from the overlaid meshes. τ can be used to change the homotopy type of the mapping.

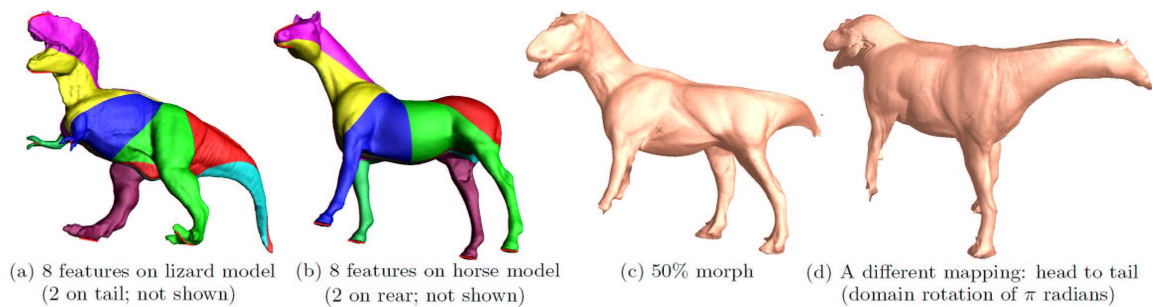


Figure 51: Surface mapping between horse and lizard. The color-coding shows the mapping of each region, guided by eight user-specified feature curves. Our topology-driven method provides mappings of different homotopy type between the two surfaces as shown in (c) and (d). We show feature curves in red.

Chapter 8

Performance Analysis

Discrete Ricci flow is a powerful tool for computing geometric structures on general surfaces. In the following, we report part of our experimental results and analyze the performance of our algorithms computing general surface geometric structures using Ricci flow from the following aspects.

8.1 Convergence

For both the Euclidean and hyperbolic cases, the discrete Ricci energies are convex. Therefore, there exists a unique global minimum. Both gradient descent and Newton's method converge to it stably. For the spherical case, the Ricci energy cannot be theoretically proven to be strictly convex. The desired metric is a critical point of the energy. In our experiments, the desired spherical metrics still can be reached efficiently using Newton's method.

8.2 Speed and Time Complexity

In the whole algorithm pipeline of computing general surface geometric structures, the most time consuming part is to compute the uniformization metrics using discrete surface Ricci flow. Figures 52 and 53 show the statistics for the computations of uniformization metrics for models with general topologies. They are all with 20k faces. The x-axis indicates the time, and the y-axis shows the maximal

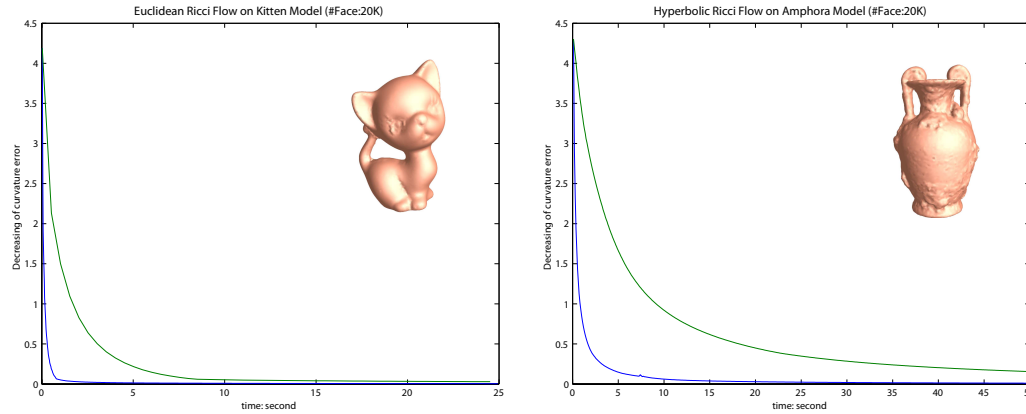


Figure 52: (a) Performance of discrete Euclidean ricci flow: blue curves are with Newton's method; green curves are with steepest descent method. (b) Performance of discrete hyperbolic ricci flow: blue curves are with Newton's method; green curves are with steepest descent method.

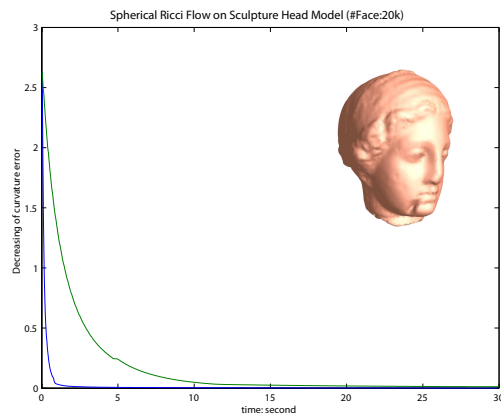


Figure 53: Performance of discrete spherical ricci flow: blue curves are with Newton's method; green curves are with steepest descent method.

curvature error. We compare the performances of the optimization of Ricci energy using Newton's method (the blue curves) and the gradient descent method (the green curves). It is obvious that Newton's method is much faster than the gradient descent method for optimizing Ricci energy. We implemented our system using C++ on a Windows platform desktop with 3.4GHz CPU Intel Xeon, 4.0G RAM using Newton's method. We report the timings to compute the desired metric in Table 9. The speed for the hyperbolic case is the slowest.








Mesh Model	# Face	genus	Bnds	Sing	Type	Time (Sec)
	20438	1	0	0	\mathbb{E}^2	3.53
	12186	0	4	0	\mathbb{E}^2	2.365
	12186	0	4	1	\mathbb{E}^2	2.528
	4117	2	0	0	\mathbb{H}^2	3.89
	20010	2	0	0	\mathbb{H}^2	9.86
	19638	3	0	0	\mathbb{H}^2	13.45
	20226	0	0	0	\mathbb{S}^2	3.76

Table 9: Time of computing discrete Ricci flow.

8.3 Conformality

Fig. 54 shows an irregular planar domain in (a) is mapped to a disk in (b). The conformality is shown with checker-board texture mapping in (c). We compute the ratio between each corner angle in (a) and in (b). The histogram of the ratios is illustrated in (d), and we can see the ratio is highly concentrated around 1.

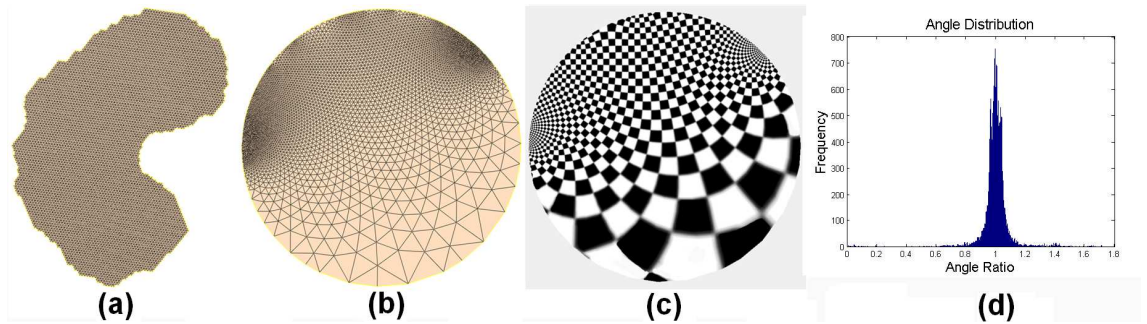


Figure 54: Conformality Testing Discrete Euclidean Ricci flow conformally maps an irregular shape in (a) to a disk in (b). The parameterization is illustrated as the check board texture mapping in (c), where all the corners of the checkers are well-preserved visually. The histogram of the angle ratio, which is defined as the ratio of the original angle value and the new one after mapping, is calculated and shown in (d). The distribution highly concentrates at 1.

8.4 Accuracy

From our experiments, we found that it is relatively easier to compute the metric, but it is more challenging to compute the layout due to the accumulation error, especially for the hyperbolic case. In order to improve the accuracy for hyperbolic flattening, we adapt the following methods.

a) Control the target metric by the maximum curvature error. We found the quality of the final embedding is strongly affected by the quality of the final metric. On the other hand, the running time depends on the maximum curvature error also. Through thorough testing, we find a good balance to set the maximum curvature error to be $1e-6$ for a mesh with $30k$ faces, then the embedding result is satisfactory and the computation speed is reasonable.

- b) Move all of the computation to the center of the Poincaré disk. Because the area distortion in the Poincaré disk is non-uniform, computations near the boundaries are highly unstable, where the area distortion goes to infinity. We use a Möbius transformation to move each current processing face to the center of the Poincaré disk, where the area distortion is close to 1 and the computation is much more stable.
- c) In flattening each face, we avoid using trigonometric functions and use more algebraic functions. For example, in order to set the third vertex position of a triangle, we compute the intersection points of two circles.
- d) Instead of flattening face by face for embedding the universal covering space, we compute the deck transformations and flatten fundamental domain by fundamental domain. This greatly reduces the accumulated error.
- e) Divide and conquer. If the mesh is big, we partition it into patches and flatten each patch, then use Möbius transformations to glue different patches together. Each edge on the cut determines a Möbius transformation; we take the average of the transformations induced by all the edges on the same cut.

We conduct a special experiment to test the hyperbolic accuracy as shown in Fig. 55. We flatten a genus two mesh in (a) with hyperbolic uniformization metric onto the Poincaré disk. We started from different seed faces marked with different colors and get different layouts of its canonical fundamental domain (its canonical fundamental generators are marked with red in both (a) and (b)), as shown in (c), (d), and (e). By using Möbius transformations given in table 10, the layouts in (d) and (e) are transformed to align perfectly with the layout in (c). Then we compute a deck transformation to glue two layouts as shown in (g). The deck transformation is computed three times using different edges on the same side as shown in (f), where three edges are colored as red, green, and blue. Each Möbius transformation is represented by two parameters: an angle θ and a center z_0 , listed in table 10. The differences among the three angles are less than $1e - 6$; the differences among three centers are also less than $1e - 6$. This shows the accuracy of our hyperbolic embedding method.

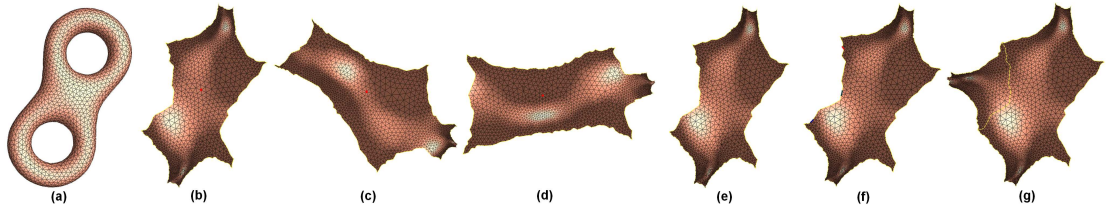


Figure 55: Accuracy Testing for Hyperbolic Layout Frame (a)-(b): a set of fundamental group generators are marked with red on a genus two eight model with $4k$ faces, and three randomly chosen seed faces are marked with red, green, and blue respectively. Frame (c)-(e) show the flattened results on the Poincaré disk using the hyperbolic uniformization metric, with different seed faces. Any two of them only differ a Möbius transformation. Frame (f): randomly chosen Edges marked with different colors can be used to compute the same deck transformation shown in (g).

Möbius Trans.	θ	Z_0
M_0	2.917271	$(-0.200690, -0.568453)$
M_1	0.567858	$(0.197063, -0.776343)$
M_2	-2.982119	$(0.187574, 0.951998)$
M_3	-2.982118	$(0.187576, 0.951999)$
M_4	-2.982113	$(0.187578, 0.951990)$

Table 10: Accuracy Testing for Hyperbolic Layout in Fig. 55. Möbius transformation M_0 moves domain in (d) to (c); M_1 moves (e) to (c); M_2 , M_3 , and M_4 are deck transformations computed from different edges on the same boundary segment.

Chapter 9

Summary and Future Research

9.1 Summary

In this work, theoretically rigorous and practically efficient methods for computing general surface geometric structures, including conformal structure, affine structure, hyperbolic structure, real projective structure, and spherical structure, are presented. The powerful tool we used is discrete surface Ricci flow. We generalized surface Ricci flow from continuous to discrete setting, and designed a series of algorithms to compute discrete surfaces Ricci flow, which includes discrete Euclidean Ricci flow, discrete hyperbolic Ricci flow, and discrete spherical Ricci flow.

We applied surface geometric structures computed from discrete surface Ricci flow to computer graphics, medical imaging, geometric modeling, and computer vision. We compute globally conformal parametrization for surfaces of general topologies, with less area distortion and control of both the number and location of singularity points; we conformally flatten colon surfaces onto plane, which enhances the navigation of virtual colonoscopy system; we design N-RoSy field on general surfaces based on flat metric induced from surfaces' conformal structure; we construct manifold spline with single singularity using surface affine structure, which achieves the theoretical minimum of the singularity number; we combine manifold spline and T-spline to polycube T-spline by building polycube map of surface which naturally induces surface affine structure; we compute shape space for general surfaces, where Surfaces are indexed and classified by their conformal

structure.

9.2 Future Research

My previous research focused on computing geometrical structures on surfaces. While studying the topological and geometric structures on three dimensional manifolds also has fundamental importance in both science and engineering. Computational algorithms for 3- manifolds can not only help topologist and geometers to investigate the complicated structures of 3-manifolds, but also have great potential for wide applications in engineering fields, including volumetric parameterizations, volumetric shape analysis, volumetric deformation, and volumetric modeling etc. Currently we are doing harmonic volumetric parameterization using Green's functions on star shapes [22] and computing discrete curvature flow for hyperbolic 3-manifolds with Complete Geodesic Boundaries [55].

9.2.1 Harmonic Volumetric Parameterization Using Green's Functions on Star Shapes

Parameterization plays a fundamental role in geometric processing, and harmonic maps have been broadly applied for surface parameterizations. A harmonic map from a topological disk to a convex planar domain is smooth, one-to-one and onto (a diffeomorphism), if the restriction of the map on the boundary is a homeomorphism, which lays down the foundation for harmonic surface parameterization. The major difficulty for generalizing harmonic parameterizations from the surface case to the volumetric case is that there is no theoretic results to ensure the harmonic maps between topological balls and convex domains to be diffeomorphisms.

We proposes a novel method to tackle this problem based on the Green's functions on star shape domains. We prove that a Green's function of a volumetric start shape has a unique critical point. Then we construct maps between volumetric star shapes in the following way: First we construct a Green's function on each of them, and compute a conformal map between their boundaries. Then we match their level sets and gradient lines. Each point is the intersection of a level set and a gradient line. This procedure induces a map between the two star shapes, which is

guaranteed to be a diffeomorphism.

In order to parameterize a general volume, we construct a volumetric polycube with similar shape, then match the boundaries of the two volumes using polycube maps. Next we decompose both volumes to star shaped segments, and use Green's functions to match the corresponding segments with consistent boundary conditions. Finally we glue the local maps together, and use volumetric harmonic map to improve the smoothness of the map along the boundaries between segments. The whole pipeline of our algorithms is illustrated in 56.

9.2.2 Discrete Curvature Flow for 3-Manifolds

A 3-manifold with complicated topology is shown in figure 58. In general, topology of 3-manifolds is extremely difficult to analyze. The perception of the topological structures of 3-manifolds is in general beyond humans intuition, because most 3-manifolds can not be realized in R^3 .

While like all surfaces embedded in the three dimensional Euclidean space have a canonical Riemannian metric, which induces constant Gaussian curvature, three dimensional manifolds also have canonical metrics, which induce constant sectional curvature. Canonical metrics on 3-manifolds are valuable for studying their topologies and have the potential for volumetric parameterization and shape matching. We have generalized discrete curvature flow for surfaces with negative Euler numbers to hyperbolic 3-manifolds with complete geodesic boundaries. The metric deforms according to the curvature, until the curvature is constant everywhere.

We apply our discrete hyperbolic curvature flow to visualize 3-manifolds, which are given by abstract tetrahedron meshes and can not be realized in R^3 . Figure 58 shows one example. This can be applied to verifying whether two 3-manifolds are homeomorphic by comparing their hyperbolic volumes in the future. If two hyperbolic 3-manifolds with geodesic boundaries are homeomorphic, then their hyperbolic volumes should be identical, independent of their triangulations, which will be one of our future directions.

In the future, I will go along the volumetric direction, generalizing discrete

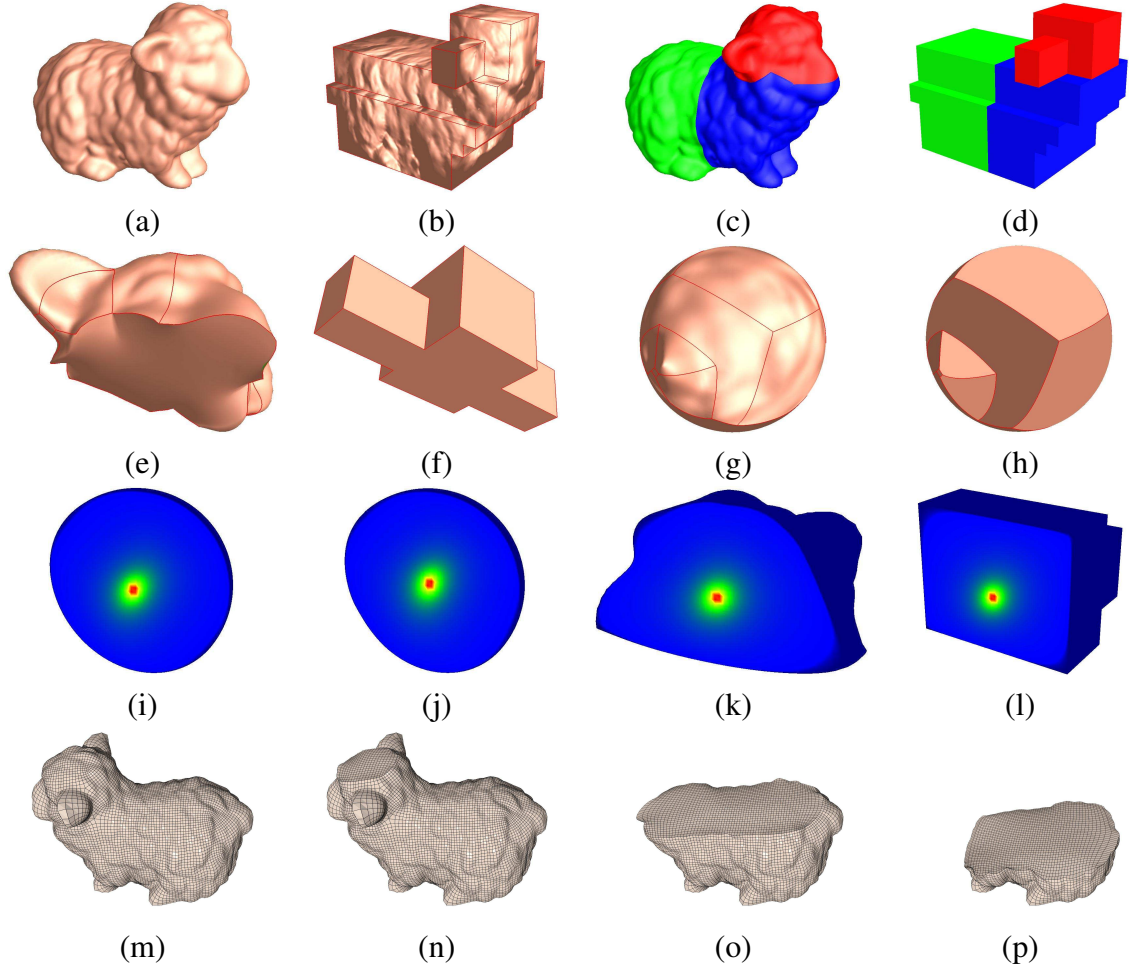


Figure 56: Algorithm pipeline. Given the genus zero sheep model, let ∂M be the boundary surface (a). We first construct a conformal polycube map $\phi_0 : \partial M \rightarrow \partial D$ (b). Then we decompose both the 3D model and polycube into three components M_i, D_i , $i = 1, 2, 3$ (see (c) and (d)). Each of M_i or D_i is a star shape, for example, the sheep head in (e) and (f). Next, we conformally map ∂M_i and ∂D_i to a sphere (see (g) and (h)). Then we compute Green's functions on the spheres. The cut views of the Green's function are shown in (i) and (j). The Green's function induces a one-to-one map between M_i and D_i as shown in (k) and (l). By gluing the segmented components together and improving the continuity along the cutting boundaries, we get the global volumetric parameterization between M and D . (m), (o) and (p) show the hexahedral remeshing results using the constructed volumetric parameterization.

2-dimensional Ricci flows to 3-dimensional volumetric curvature flows, and apply them in volumetric shape mapping, and volumetric parametrization for general

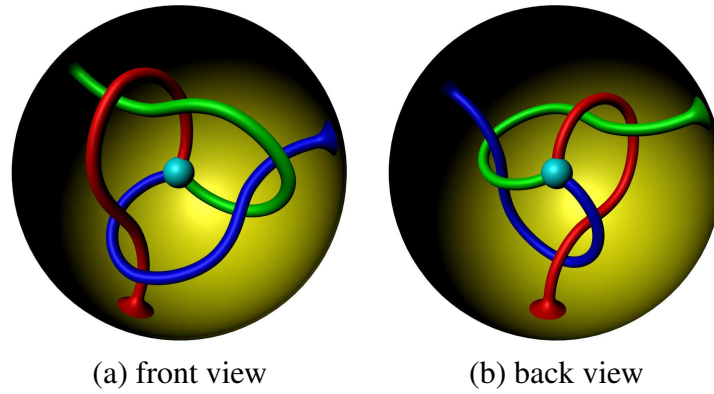


Figure 57: The boundary surface of a 3-manifold, Thurston's knotted Y. The volume can be treated as a solid ball with three entangled tunnels.

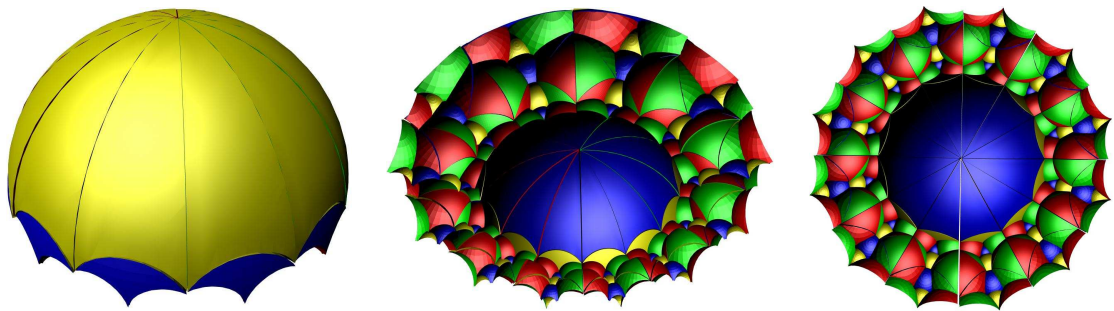


Figure 58: Embedding the 3-manifold periodically in the hyperbolic space H^3 with different viewpoints.

shapes in computer graphics; volumetric medical data registration and fusion in medical imaging; volumetric shape registration, analysis, and retrieval in computer vision; volumetric spline in geometric modeling.

Bibliography

- [1] P. Alliez, M. Meyer, and M. Desbrun. Interactive geometry remeshing. *SIG-GRAPH 02*, pages 347–354, 2002.
- [2] S. Angenent, S. Haker, A. Tannenbaum, and R. Kikinis. Conformal geometry and brain flattening. *MICCAI*, pages 271–278, 1999.
- [3] M. Ben-Chen, C. Gotsman, and G. Bunin. Conformal flattening by curvature prescription and metric scaling. *Comp. Graph. Forum*, 27(2), 2008.
- [4] J. P. Benzècri. Variétés localement affines. *Sem. Topologie et Gèom. Diff., Ch. Ehresmann*, (7):215–223, 1959.
- [5] P. Buser. Geometry and spectra of compact riemann surfaces. *Birkhauser*, 1992.
- [6] C. Carner, M. Jin, X. Gu, and H. Qin. Topology-driven surface mappings with robust feature alignment. *IEEE Visualization*, pages 543–550, 2005.
- [7] J. Chen and Y. Han. Shortest paths on a polyhedron; part i: computing shortest paths. *Int. J. Comput. Geom. and Appl.*, 6(2):127–144, 1996.
- [8] B. Chow and F. Luo. Combinatorial ricci flows on surfaces. *Journal Differential Geometry*, 63(1):97–129, 2003.
- [9] M. Desbrun, M. Meyer, and P. Alliez. Intrinsic parameterizations of surface meshes. *Computer Graphics Forum (Proc. Eurographics 2002)*, 21(3):209–218, 2002.

- [10] H. Ferguson, A. P. Rockwood, and J. Cox. Topological design of sculptured surfaces. *SIGGRAPH*, pages 149–156, 1992.
- [11] M. S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, 20(1):19–27, 2003.
- [12] S. J. Gortler, C. Gotsman, and D. Thurston. Discrete one-forms on meshes and applications to 3D mesh parameterization. *Computer Aided Geometric Design*, 23(2):83–112, 2005.
- [13] C. Grimm and J. F. Hughes. Parameterizing n-holed tori. *IMA Conference on the Mathematics of Surfaces*, pages 14–29, 2003.
- [14] X. Gu, Y. He, M. Jin, F. Luo, and H. Qin. Manifold splines with single extraordinary point. *Computer Aided Design*, 40(6):676–690, 2008.
- [15] X. Gu, Y. He, and H. Qin. Manifold splines. *Graphical Models*, 68(3):237–254, 2006.
- [16] X. Gu and B. C. Vemuri. Matching 3d shapes using 2d conformal representations. *MICCAI (1)*, pages 771–780, 2004.
- [17] X. Gu, Y. Wang, T. F. Chan, P. M. Thompson, and S.-T. Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Trans. Med. Imaging*, 23(8):949–958, 2004.
- [18] X. Gu and S.-T. Yau. Computing conformal structures of surfaces. *Communications in Information and Systems*, 2(2):121–146, December 2002.
- [19] X. Gu and S.-T. Yau. Global conformal surface parameterization. *ACM Symposium on Geometry Processing*, pages 127–137, 2003.
- [20] S. Haker, S. Angenent, A. Tannenbaum, R. Kikinis, G. Sapiro, and M. Halle. Conformal surface parameterization for texture mapping. *IEEE Transactions on Visualization and Computer Graphics*, 6(2):181C189, 2000.
- [21] R. S. Hamilton. Three-manifolds with positive ricci curvature. *J. Diff. Geom.*, 17:255–306, 1982.

- [22] Y. He, X. Ying, M. Jin, F. Luo, and X. Gu. Harmonic volumetric parameterization using greens functions on star shapes. *Symposium on Geometry Processing*, 2008.
- [23] W. Hong, X. Gu, F. Qiu, M. Jin, and A. Kaufman. Conformal virtual colon flattening. *SPM '06: Proceedings of the 2006 ACM symposium on Solid and physical modeling*, pages 85–93, 2006.
- [24] M. Jin, J. Kim, F. Luo, and X. Gu. Discrete surface ricci flow. *IEEE Transaction on Visualization and Computer Graphics*, 14(5), 2008.
- [25] M. Jin, F. Luo, and X. Gu. Computing surface hyperbolic structure and real projective structure. *ACM Symposium on Solid and Physics Modeling*, pages 105–116, 2006.
- [26] M. Jin, F. Luo, S.-T. Yau, and X. Gu. Computing geodesic spectra of surfaces. *ACM Symposium on Solid and Physics Modeling*, pages 387–393, 2007.
- [27] M. Jin, Y. Wang, S.-T. Yau, and X. Gu. Optimal global conformal surface parameterization. *IEEE Visualization 2004*, pages 267–274, 2004.
- [28] M. Jin, W. Zeng, F. Luo, and X. Gu. Computing teichmüller shape space. *IEEE Transaction on Visualization and Computer Graphics, under minor revision*, 2008.
- [29] J.R.Munkres. Elements of algebraic topology. *Addison-Wesley Co.*, 1984.
- [30] J.W.Milnor. On the existence of a connection with curvature zero. *Comm. Math. Helv.*, 7(2):229–233, 1958.
- [31] B. KANEVA and J. OROURKE. An implementation of chen and hans shortest paths algorithm. *Proc. of the 12th Canadian Conf. on Comput. Geom.*, pages 139–146, 2000.
- [32] M. Kaplan and E. Cohen. Computer generated celtic design. *Proceedings of the 14th Eurographics Workshop on Rendering Techniques*, pages 2–19, 2003.

- [33] S. Kapoor. Efficient computation of geodesic shortest paths. *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 770–779, 1999.
- [34] L. Kharevych, B. Springborn, and P. Schröder. Discrete conformal mappings via circle patterns. *ACM Transactions on Graphics*, 25(2):412–438, 2006.
- [35] R. KIMMEL and J. A. SETHIAN. Computing geodesic paths on manifolds. *Proc. of National Academy of Sci.*, 95(15):8431–8435, 1998.
- [36] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. *SIGGRAPH*, pages 362–371, 2002.
- [37] A. K. D. B. T. H. Lichan Hong, Shigeru Muraki. Virtual voyage: Interactive navigation in the human colon. *SIGGRAPH 1997*, page 27C34, 1997.
- [38] F. Luo. Geodesic length functions and teichmüller spaces. *J. DIFFERENTIAL GEOMETRY*, 48:275, 1998.
- [39] J. Maillot, H. Yahia, and A. Verroust. Interactive texture mapping. *Computer Graphics (Proceedings of SIGGRAPH 93)*, pages 27–34, 1993.
- [40] T. S. Mika Seppala. Geometry of riemann surfaces and teichmüller spaces. *North-Holland Math Stud*, 1992.
- [41] J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou. The discrete geodesic problem. *SIAM J. Comput.*, 16(4):647–668, 1987.
- [42] J. Palacios and E. Zhang. Rotational symmetry field design on surfaces. *ACM Trans. Graph.*, 26(3):55, 2007.
- [43] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.
- [44] N. Ray, B. Vallet, W.-C. Li, and B. Levy. N-symmetry direction field design. *ACM Trans. Graph.*, *accepted pending revisions*, to appear.

- [45] A. Sheffer and E. de Sturler. Parameterization of faced surfaces for meshing using angle based flattening. *Engineering with Computers*, 17(3):326–337, 2001.
- [46] A. Sheffer, B. Lévy, M. Mogilnitsky, and A. Bogomyakov. ABF++: Fast and robust angle based flattening. *ACM Transactions on Graphics*, 24(2):311–330, 2005.
- [47] V. Surazhsky, T. Surazhsky, D. Kirsanov, S. J. Gortler, and H. Hoppe. Fast exact and approximate geodesics on meshes. *ACM Trans. Graph.*, 24(3):553–560, 2005.
- [48] W. P. Thurston. Geometry and topology of three-manifolds. *Princeton lecture notes*, 1976.
- [49] W. P. Thurston. Three-dimensional geometry and topology. *Princeton University Press*, 1997.
- [50] Y. Tong, P. Alliez, D. Cohen-Steiner, and M. Desbrun. Designing quadrangulations with discrete harmonic forms. *Symposium on Geometry Processing*, pages 201–210, 2006.
- [51] J. Wallner and H. Pottmann. Spline orbifolds. *Curves and Surfaces with Applications in CAGD*, pages 445–464, 1997.
- [52] H. Wang, M. Jin, Y. He, X. Gu, and H. Qin. User-controllable polycube map for manifold spline construction. *ACM Symposium on Solid and Physics Modeling*, pages 397–404, 2008.
- [53] L. Wang, X. Gu, K. Mueller, and S.-T. Yau. Uniform texture synthesis and texture mapping using global parameterization. *The Visual Computer (Special issue of Pacific Graphics '05)*, pages 801–810, 2005.
- [54] S. H. Weitraub. Differential forms: A complement to vector calculus. *Academic Press*, 2007.

- [55] X. Yin, M. Jin, F. Luo, and X. Gu. Discrete curvature flow for hyperbolic 3-manifolds with complete geodesic boundaries. *Symposium on Geometry Processing*, 2008.
- [56] R. Zayer, C. Rössl, and H.-P. Seidel. Setting the boundary free: A composite approach to surface parameterization. *Eurographics Symposium on Geometry Processing*, pages 91–100, 2005.