Computer-Aided Design 111 (2019) 1–12

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

Curvature adaptive surface remeshing by sampling normal cycle☆

Kehua Su [a], Na Lei [b,c,g,*], Wei Chen [b], Li Cui [d], Hang Si [f], Shikui Chen [h], Xianfeng Gu [e,g]

[a] School of Computer Science, Wuhan University, Wuhan, 430072, China
[b] DUT-RU ISE, Dalian University of Technology, Dalian, 116620, China
[c] Key Lab. for Ubiquitous Network and Service Software of Liaoning Province, Dalian 116620, China
[d] School of Mathematical Science, Beijing Normal University, Beijing, 100084, China
[e] Computer Science Department, Stony Brook University, Stony Brook, NY, 11794, USA
[f] Numerical Mathematics and Scientific Computing Weierstrass Institute for Applied Analysis and Stochastics, Berlin, 10117, Germany
[g] Beijing Advanced Innovation Center for Imaging Technology, Beijing, 100048, China
[h] Department of Mechanical Engineering, Stony Brook University, Stony Brook, NY, 11794, USA

ARTICLE INFO

Article history:
Received 1 October 2017
Accepted 28 January 2019

Keywords:
Surface remeshing
Normal cycle
Dynamic Ricci flow
Optimal transport
Conformal parameterization
Area-preserving parameterization

ABSTRACT

Surface meshing plays a fundamental important role in Visualization and Computer Graphics, which produces discrete meshes to approximate a smooth surface. Many geometric processing tasks heavily depend on the qualities of the meshes, especially the convergence in terms of topology, position, Riemannian metric, differential operators and curvature measures.

Normal cycle theory points out that in order to guarantee the convergence of curvature measures, the discrete meshes are required to approximate not only the smooth surface itself, but also the normal cycle of the surface. This theory inspires the development of the remeshing method based on conformal parameterization and planar Delaunay refinement, which uniformly samples the smooth surface, and produces Delaunay triangulations with bounded minimal corner angles. This method ensures the Hausdorff distances between the normal cycles of the resulting meshes and the smooth normal cycle converges to 0, the discrete Gaussian curvature and mean curvature measures of the resulting meshes converge to their counter parts on the smooth surface.

In the current work, the conformal parameterization based remeshing algorithm is further improved to speed up the curvature convergence. Instead of uniformly sampling the surface itself, the novel algorithm samples the normal cycle of the surface. The algorithm pipeline is as follows: first, two parameterizations are constructed, one is the surface conformal parameterization based on dynamic Ricci flow, the other is the normal cycle area-preserving parameterization based on optimal mass transportation; second, the normal cycle parameterization is uniformly sampled; third, the Delaunay refinement mesh generation is carried out on the surface conformal parameterization. The produced meshes can be proven to converge to the smooth surface in terms of curvature measures.

Experimental results demonstrate the efficiency and efficacy of proposed algorithm, the convergence speeds of the curvatures are prominently faster than those of conventional methods.

1. Introduction

Surface meshing and remeshing are of the fundamental importance in visualization and computer graphics, as well as many engineering and medicine fields, including geometric modeling, digital geometry processing and medical imaging. Typically, surface meshing finds a set of sample points on the surface with a curved triangulation, then approximates each face by a Euclidean triangle in $\mathbb{R}^3$, thereby approximating the underlying smooth surface by a polyhedral triangular surface, which is called a triangular mesh.

1.1. Different levels of convergence

Many downstream geometric processing algorithms, such as editing, parameterization, analyzing, registration, deformation, storing and transmission, fundamentally rely on the mesh qualities. It is required that the discrete meshes converge to the smooth surface with different levels of accuracies. Generally speaking, there are three major levels of convergence:

(1) Topological convergence, the discrete meshes and the smooth surface are homeomorphic.

☆ This paper has been recommended for acceptance by S.J. Owen.

* Corresponding author at: DUT-RU ISE, Dalian University of Technology, Dalian, 116620, China.
E-mail address: nalei@dlut.edu.cn (N. Lei).

https://doi.org/10.1016/j.cad.2019.01.004
0010-4485/

(2) Positional convergence, the Hausdorff distance between the discrete mesh and the surface limits to zero.
(3) Curvature measure convergence, the Gaussian and mean curvature measures of the discrete mesh converge to the counter parts on the smooth surface. This is equivalent to the convergence in terms of normal, Riemannian metric (including area element) and the Laplace–Beltrami operator.

The lower level convergence does not imply the higher level convergence. For example, the famous Schwartz's lantern [1] has the positional convergence, but not the area convergence. In order to guarantee the curvature measure convergence, one needs the guidance from normal cycle theory [1–3].

1.2. Normal cycle

Normal cycle theory [1–3] unifies the concepts of curvatures for both smooth surfaces and discrete meshes, and quantitatively measures the approximation accuracy, convergence rate of the curvature measures. Essentially, normal cycles that close in the Hausdorff distance induce similar curvature measures.

Suppose S is a C^2 smooth convex surface, embedded in the three dimensional Euclidean space $\mathbb{E}^3$. The Gaussian curvature is well defined. Assume $p \in S$ is a point on the surface, the normal at p is denoted as $\mathbf{n}(p)$. The *offset* surface of S is defined as

$$N(S) := \{p + \mathbf{n}(p) | p \in S\}. \tag{1}$$

Since S is convex, the offset surface $N(S)$ is also convex, embedded in $\mathbb{E}^3$.

Similarly, suppose M is a convex mesh, the discrete Gaussian curvature can be defined as the angle deficit at the vertices. A *supporting plane* of M is a plane that has both of the following properties:

- M is entirely contained in one of the two closed half-spaces bounded by the plane.
- M has at least one point on the plane.

The *normal cone* at the point p is defined as the set of all normals to the supporting planes through p, and denoted as $NC(p)$. Similar to Eq. (1), the *discrete offset surface* of M is defined similarly,

$$N(M) := \{p + \mathbf{n} | p \in M, \quad \mathbf{n} \in NC(p)\} \tag{2}$$

Since M is convex, $N(M)$ is convex and embedded in $\mathbb{E}^3$.

Now suppose we want to use a sequence of discrete meshes $\{M_n\}$ to approximate the smooth surface S, even if the Hausdorff distance between M_n and S converges to 0,

$$\lim_{n\to\infty} d_H(M_n, S) = 0,$$

where $d_H(\cdot, \cdot)$ represents the Hausdorff distance, the discrete Gaussian curvature measure, or the areas, geodesics, Laplace–Beltrami spectra of M_n may not converge to the corresponding geometric quantity of S. In contrast, if the Hausdorff distance between the discrete and the smooth offset surfaces converges to 0,

$$\lim_{n\to\infty} d_H(N(M_n), N(S)) = 0,$$

then by the theory of Steiner (chapter 16, [1]), the discrete curvature measure of M_n converges to the curvature measure of the smooth surface. Furthermore, the area element, geodesics, and Laplace–Beltrami spectra of M_n converge to those of the smooth surface as well. Therefore, we should consider the Hausdorff distance between $N(S)$ and $N(M)$ to guarantee the higher order convergence.

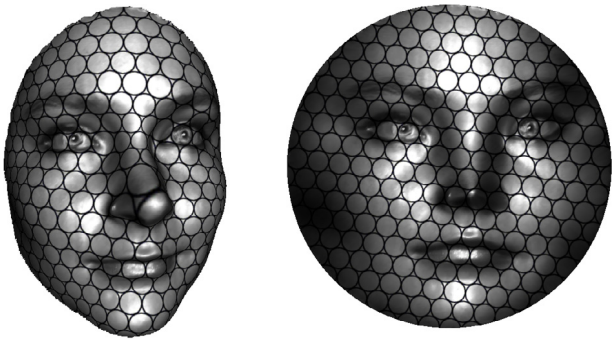
Unfortunately, when S is non-convex, its offset surface $N(S)$ may have self-intersection, then it is difficult to measure the Hausdorff distance between the offset surfaces. In order to avoid the self-intersection, the offset surface is generalized to the concept of *normal cycle*. The key difference is that the offset surface is embedded in $\mathbb{E}^3$, whereas the normal cycle is embedded in the product space $\mathbb{E}^3 \times \mathbb{S}^2$. We use $N(S)$ to represent the normal cycle. Formally, the normal cycle for S is defined as

$$N(S) := \{(p, \mathbf{n}(p)) | p \in S\}. \tag{3}$$

and the discrete normal cycle for M is defined

$$N(M) := \{(p, \mathbf{n}) | p \in S, \mathbf{n} \in NC(p)\}. \tag{4}$$

The normal cycle theory [1] shows that in order to improve the curvature convergence speed, one needs to improve the approximation accuracy to the normal cycle, instead of that to the surface itself.



(a) Human facial surface (b) Conformal parameterization

Fig. 1. Conformal parameterization preserves infinitesimal circles, therefore preserves Delaunay triangulations.

1.3. Sampling strategies

Based upon normal cycle theory, for the purpose of curvature convergence, the meshing algorithm needs to meet two criteria, one is the angle criteria, the other one is the sampling density criteria:

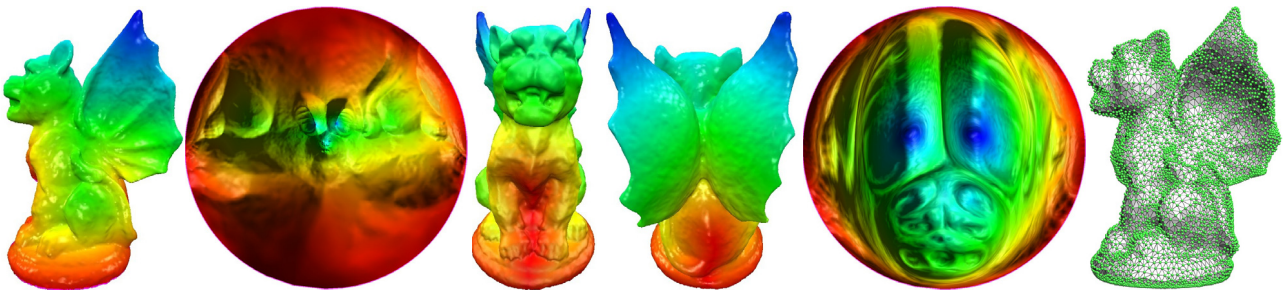
(1) Angle criteria: there is a positive constant $c > 0$, such that the minimal corner angles of all meshes are greater than c.
(2) Density criteria: suppose S is a smooth surface embedded in $\mathbb{E}^3$, S_ε is an ε-sample of S, namely, for each point $p \in S$, the ball $B(p, \varepsilon\, lfs(p))$ contains at least one sample point in S_ε, where $lfs(p)$ denotes the local feature size [4] of S at the point p.

In order to satisfy these two criteria, different algorithms choose different strategies. Roughly speaking, there are three major types of algorithms:

1. Sampling the Surface in $\mathbb{E}^3$ The method [4] directly samples the surface in $\mathbb{E}^3$, and computes the Delaunay tetrahedralization of the interior volume. The restriction of the Tetrahedral triangulation on the surface gives the meshing result.

2. Sampling the parameter domain Because the conformal mapping preserves infinitesimal circles, Delaunay triangulations have the empty-circle property, therefore conformal mapping preserves Delaunay triangulations (see Fig. 1). Comparing to volumetric Delaunay triangulation, planar Delaunay triangulation is much more efficient.

The algorithms introduced in [5] and [3] compute surface conformal parameterization, then use the conformal factor as the grading factor, samples on the parameter domain. The Delaunay triangulation is constructed on the parameter domain, and induces the surface triangulation. This method produces uniform samplings



(a) side view (b) CFP parameterization (c) front view (d) back view (e) CAP parameterization (f) remeshed result

Fig. 2. Conformal parameterization (CFP), Curvature adaptive parameterization (CAP) and Remeshing result for Gargoyle model. The original model is 68k vertices and the remeshed mesh is 4k vertices.

on the surface, and the resulting Delaunay triangles are uniformly sized.

3. Sampling the Normal Cycle In the current work, we propose a novel algorithm by sampling the normal cycle of the original surface directly. The normal cycle is defined in the 6 dimensional space, it is highly expensive to sample it using high dimensional Delaunay triangulation directly. Therefore, we utilize parameterization to map the two-dimensional normal cycle embedded in the 6-dimensional space to a planar domain, and sample the $2D$ parameter domain. In order to control the sampling density, we utilize the area-preserving parameterization from the normal cycle to the planar domain. The area-preserving parameterization only requires the Riemannian metric of the normal cycle, which can be easily calculated as

$$ds_N^2 := \langle d\mathbf{p}, d\mathbf{p}\rangle + \langle d\mathbf{n}, d\mathbf{n}\rangle \tag{5}$$

where $\mathbf{p}$ is the position vector of the point, $\mathbf{n}$ is the normal vector.

We uniformly sample the area-preserving parameter domain of the normal cycle, then map the samples to the conformal parameter domain of the surface, and calculate the Delaunay triangulation on the conformal parameter domain, which induces a Delaunay geodesic triangulation on the original surface (see Fig. 2).

1.4. Contributions

This work introduces a novel curvature adaptive remeshing algorithm based on normal cycle theory, which has the following merits:

(1) Most conventional methods uniformly sample the input surface, in contrast, the proposed method samples the normal cycle of the surface, this greatly improves the speed of curvature measure convergence.
(2) This method adopts the dynamic discrete Yamabe flow method for conformal parameterization. Comparing to conventional discrete Ricci flow method, dynamic Yamabe flow is very robust to the mesh quality, and the solution existence has theoretic guarantee.
(3) The proposed method has solid theoretic foundations, the convergence of the curvature measures of the result meshes can be easily proven.
(4) The discrete optimal mass transportation map offers great flexibility for sampling, for example, it supports different levels of emphasis of the curvature measures and region of interests.

2. Previous work

2.1. Surface remeshing

Surface remeshing has been an active research subject for nearly two decades, see a nice survey given by Alliez et al. [6]. Most of the early methods work either in a 2d parameterization space [5,7] or directly in the 3d space [8], and their focus is to create isotropic 3d surface meshes. These methods usually provide no guarantee on the convergence between their resulting meshes and the approximating surfaces.

Later on, theoretical methods based on *restricted Delaunay triangulation* (RDT) have been developed. Edelsbrunner and Shah [9] introduced Delaunay triangulation of topological spaces, and proved the reconstructed simplicial complex is homeomorphic to the sampled topological space; Cheng, Dey and Levine used Delaunay refinement for the meshing purpose in [10], and proved the method is topology preserving. Amenta and Bern proposed surface reconstruction by Voronoi filtering in [11], developed the concept of ε-sample, and showed the convergence in terms of both *topology* and *normal*; Morvan and Thibert showed the restricted Delaunay triangulation based on ε-sampling guarantees the convergence of the surface [12]. Cohen-Steiner and Morvan proved reconstruction results based on ε-sampling and DRT ensures the convergence of curvature. Boissonnat and Oudot generalized the ε-sample to loose ε-sample in [13], and proved the convergence in terms of topology, normal, area and curvature.

Centroidal Voronoi Tessellation (CVT) [14–18] has been developed. In practice, CVT method can achieve high quality meshing results. These methods are able to guarantee the topology, but the formal proofs for the convergence on geometry and curvature have not been reported. Many recent methods have been developed for creating anisotropic surface meshes [19–25]. In particular, the methods in [24,25] will result in a curvature-adapted surface mesh by using the normal (Gaussian map) of the surface. However, for discrete surfaces, they provide no proof for curvature convergence.

2.2. Meshing

The Delaunay refinement algorithms were originally designed for meshing planar domains, and were later generalized for meshing surfaces and volumes. Chew's first algorithm [26] splits any triangle whose circumradius is greater than the prescribed shortest edge length parameter e and hence generates triangulation of uniform density and with no angle smaller than 30°. But the number of triangles produced is not optimal. Chew's second algorithm [27] splits any triangle whose circumradius-to-shortest-edge ratio is greater than one, and hence in practice produces grade mesh. Similar split criterion was used in Ruppert's algorithm [28], which has the theoretical guarantee of the minimal angle of no less than 20.7°. Shewchuk [29] unified the pioneering mesh generation algorithms of L. Paul Chew [26] and Jim Ruppert [28], improved the algorithms in several minor ways, and helped to solve the difficult problem of meshing non-manifold domains with small angles. Dey et al. developed a series of algorithms for surface meshing and remeshing based on volumetric Delaunay refinement [30–32], which belong to the approaches in the first category. We refer readers to [33] for full details.

2.3. Ricci Flow

Ricci flow conformally deforms the Riemannian metrics, such that during the flow, the infinitesimal circles are preserved. This phenomenon inspired Thurston to develop the circle packing method. In his work on constructing hyperbolic metrics on 3-manifolds, Thurston [34] studied a Euclidean (or a hyperbolic) circle packing on a triangulated closed surface with prescribed intersection angles. Thurston conjectured that the discrete conformal mapping based on circle packing converges to the smooth Riemann mapping when the discrete tessellation becomes finer and finer. Thurston's conjecture has been proven by Rodin and Sullivan [35]. He and Schramm gave another proof in 1996 [36]. Chow and Luo established the intrinsic connection between circle packing and surface Ricci flow [37].

The rigidity for classical circle packing was proved by Thurston [34], Marden-Rodin [38], Colin de Verdiére [39], Chow–Luo [37], Stephenson [40], and He [36].

Bowers–Stephenson [41] introduced inversive distance circle packing which generalizes Andreev–Thurston's intersection angle circle packing. See Stephenson [40] for more information. Guo gave a proof for local rigidity [42] of inversive distance circle packing. Luo gave a proof for global rigidity in [43].

Luo introduced and studied the combinatorial Yamabe problem for piecewise flat metrics on triangulated surfaces [44] . Springborn, Schröder and Pinkall [45] considered this combinatorial conformal change of piecewise flat metrics and found an explicit formula of the energy function. Glickenstein [46,47] studied the combinatorial Yamabe flow on 3-dimensional piecewise flat manifolds. Bobenko–Pinkall–Springborn introduced a geometric interpretation to Euclidean and hyperbolic Yamabe flow using the volume of generalized hyperbolic tetrahedron in [48]. Combinatorial Yamabe flow on hyperbolic surfaces with boundary has been studied by Guo in [49]. The existence of the solution to Yamabe flow with topological surgeries has been proved recently in [50] and [51].

The Euclidean virtual radius circle packing first appeared in [52]. The hyperbolic and spherical virtual radius circle packing are introduced in [53].

The Euclidean mixed type circle packing appeared in [52] and Glickenstein's talk [54]. This work introduces hyperbolic and spherical mixed type schemes.

Glickenstein [55] set the theory of combinatorial Yamabe flow of piecewise flat metric in a broader context including the theory of circle packing on surfaces. [53] focused on the hyperbolic and spherical unified frameworks.

The variational approach to circle packing was first introduced by Colin de Verdiére [39]. Since then, many works on variational principles on circle packing or circle pattern have appeared. For example, see Brägger [56], Rivin [57], Leibon [58], Chow–Luo [37], Bobenko–Springborn [59], Guo–Luo [60], and Springborn [61]. Variational principles for polyhedral surfaces including the topic of circle packing were studied systematically in Luo [62]. Many energy functions are derived from the cosine law and its derivative. Tangent circle packing is generalized to tangent circle packing with a family of discrete curvature. For exposition of this work, see also Luo–Gu–Dai [63].

Recently, Gu et al. established discrete uniformization theorem based on Euclidean [50] and hyperbolic [51] Yamabe flow. In a series of papers on developing discrete uniformization theorem [64–66] and [67], Sa'ar Hersonsky proved several important theorems based on discrete harmonic maps and cellular decompositions. His approach is complementary to the work mentioned above.

2.4. Convergence

Amenta et al. [4] derived an algorithm of the reconstruction of surfaces from unorganized sample points in $\mathbb{R}^3$ which is based on the three-dimensional Voronoi diagram. Assuming given good sampling, they proved the output of the algorithm is guaranteed to be topologically correct and convergent to the original surface as the sampling density increases. This is the first work for meshing with a provable guarantee.

Cohen-Steiner and Morvan [2] derived a simple and new definition of the curvature tensor for polyhedral surfaces building upon the theory of normal cycles, which yields an efficient and reliable curvature estimation algorithm. Moreover, they bound the difference between the estimated curvature and the one of the smooth surface in the case of restricted Delaunay triangulations.

Morvan and Thibert [68] compared the normal vector field of a smooth surface S with the normal vector field of another surface differentiable almost everywhere. They gave an upper bound on angles between the normals of S and the normals of a triangulation T close to S.

Suppose the triangulation is obtained from a sampling of a smooth parametric surface, Xu [69] showed theoretically that the approximation has quadratic convergence rate if the surface sampling satisfies the so-called parallelogram criterion.

Hildebrandt, Polthier and Wardetzky [70] studied the convergence of polyhedral surfaces and their discrete geometric properties to smooth surfaces embedded in Euclidean 3-space. They showed that convergence of the following properties is equivalent under the assumption of convergence of surfaces in Hausdorff distance: surface normals, surface area, metric tensors, and Laplace–Beltrami operators.

Li et al. [3] focused on the curvature measure convergence for the conformal parameterization based Delaunay refinement algorithms. They gave explicit estimates for the Hausdorff distance, the normal deviation, and the differences in curvature measures between the surface and the mesh.

3. Theoretic background

In this section, we briefly introduce the theoretic foundation of our framework. We refer readers to [52] and [71] for detailed treatments.

3.1. Dynamic discrete surface Ricci flow

In practice, smooth surfaces are approximated by discrete triangle meshes. A triangle mesh is denoted as $M = (V, E, F)$, where V, E and F represent vertex, edge and face sets, respectively. A discrete Riemannian metric is formulated as an edge length function $l : E \to \mathbb{R}^+$, which satisfies the triangle inequality on each face. The corner angles are determined by the cosine law:

$$\theta_i = \cos^{-1} \frac{l_j^2 + l_k^2 - l_i^2}{2 l_j l_k} \tag{6}$$

where the vertices are v_i, v_j and v_k, the length of the edge against v_i is l_i, the corner angle at the vertex v_i is θ_i. The discrete Gaussian curvature for an interior vertex is defined as 2π minus the sum of the surrounding corner angles, while the Gaussian curvature for a boundary vertex is computed by π minus the sum of the surrounding corner angles:

$$K(v_i) = \begin{cases} 2\pi - \Sigma_{f_{ijk} \in F} \theta_i^{jk}, & v_i \notin \partial M \\ \pi - \Sigma_{f_{ijk} \in F} \theta_i^{jk}, & v_i \in \partial M \end{cases} \tag{7}$$

where θ_i^{jk} denotes the corner angle adjacent to the vertex v_i in the face f_{ijk}, and ∂M represents the boundary of the mesh M.

According to the Gauss–Bonnet theorem, the total curvature is a topological invariant,

$$\sum_{v_i \in V} K_i = 2\pi \chi(M), \tag{8}$$

where $\chi(M)$ is the Euler characteristic number of M. We define a discrete conformal factor $\mathbf{u}$ on vertices as $\mathbf{u} : V \to \mathbb{R}$. Suppose e_{ij} is an edge which has end vertices v_i and v_j, and d_{ij} is the edge length of e_{ij} induced by the Euclidean metric. Then the edge length l_{ij} in the discrete Ricci flow is defined as:

$$l_{ij} = \exp(u_i) d_{ij} \exp(u_j). \tag{9}$$

Let K_i be the discrete Gaussian curvature defined on vertex v_i, and $\bar{K}_i$ be the target curvature. The discrete Euclidean (and Hyperbolic) Yamabe flow is defined as:

$$\frac{du_i}{dt} = \bar{K}_i - K_i. \tag{10}$$

The following theorem guarantees the existence of the solution,

Theorem 1 (*Discrete Uniformization [50]*)**.** *If the target curvature satisfies the Gauss–Bonnet condition and $\bar{K}_i < 2\pi$ holds for each vertex v_i, then the solution to the dynamic discrete Ricci flow exists and is unique up to a constant.*

Moreover, this solution is the unique optimal point of the following convex discrete Ricci energy:

$$E(\mathbf{u}) = \int_0^{\mathbf{u}} \sum_{i=1}^{n} (\bar{K}_i - K_i) du_i, \quad \mathbf{u} = (u_1, u_2, \ldots, u_n). \tag{11}$$

In practice, we employ Newton's method to optimize the Ricci energy. The gradient of the Ricci energy is the curvature difference,

$$\nabla E(\mathbf{u}) = (\bar{K}_i - K_i)^T, \tag{12}$$

The Hessian matrix of the Ricci energy is given as follows:

$$\frac{\partial^2 E(\mathbf{u})}{\partial u_i \partial u_j} = \begin{cases} w_{ij}, & i \neq j \\ -\Sigma_k w_{ik}, & i = j \end{cases} \tag{13}$$

where w_{ij} is the cotangent edge weight of e_{ij}:

$$w_{ij} = \cot \theta_k + \cot \theta_l, \tag{14}$$

where θ_k and θ_l are two corner angles against the edge e_{ij}. During the flow, the triangulation is dynamically updated by edge swapping, such that the triangulation is Delaunay with respect to the current Riemannian metric, namely, the cotangent edge weight w_{ij} for each edge e_{ij} is always non-negative during the flow, therefore the Hessian matrix is always positive definite in the subspace $\sum_i u_i = 0$, namely the Ricci energy is convex. Dynamic Ricci flow is much more robust than conventional methods by preserving the Delaunay triangulation.

3.2. Discrete optimal mass transportation

Let X and Y be domains in the Euclidean space $\mathbb{R}^n$. Two probability measures μ and ν are given respectively with equal total measures, $\int_X \mu = \int_Y \nu$. A map $T : X \to Y$ is *measure preserving* if for any measurable set $B \subset Y$, the following condition holds: $\int_{T^{-1}(B)} \mu = \int_B \nu$. If T minimizes the following *transportation cost*,

$$E(T) := \int_X |x - T(x)|^2 d\mu(x),$$

then T is called an *optimal mass transportation map*.

In practice, we formulate the optimal transportation problem in the discrete setting by sampling the target domain into a discrete point set. Suppose μ has a compact support on X, define $\Omega = \mathrm{Supp}\ \mu = \{x \in X | \mu(x) > 0\}$, and assume Ω is a convex domain in X. The space Y is discretized into $Y = \{y_1, y_2, \ldots, y_k\}$ with Dirac measure $\nu = \sum_{i=1}^k \nu_i \delta(y - y_i)$.

We define a *height vector* $\mathbf{h} = (h_1, h_2, \ldots, h_k) \in \mathbb{R}^k$, consisting of k real numbers. For each $y_i \in Y$, we construct a hyper-plane defined on X:

$$\pi_i(\mathbf{h}) : \langle x, y_i \rangle + h_i = 0, \tag{15}$$

where $\langle , \rangle$ is the inner product in $\mathbb{R}^n$. Define a piece-wise linear function:

$$u_{\mathbf{h}}(x) = \max_{1 \leq i \leq k} \{\langle x, y_i \rangle + h_i\}, \tag{16}$$

then $u_{\mathbf{h}}$ is a convex function. We denote its graph by $\mathcal{E}(\mathbf{h})$, which is an infinite convex polyhedron with supporting planes $\pi_i(\mathbf{h})$. Namely, $\mathcal{E}(\mathbf{h})$ is the *upper envelope* of the planes $\{\pi_i(\mathbf{h})\}$. The projection of $\mathcal{E}(\mathbf{h})$ induces a *power cell decomposition* of Ω, denoted as $\mathcal{V}(\mathbf{h})$,

$$\Omega = \bigcup_{i=1}^{k} W_i(\mathbf{h}), \ W_i(\mathbf{h}) := \{x \in X | u_{\mathbf{h}}(x) = \langle x, y_i \rangle + h_i\} \cap \Omega. \tag{17}$$

The measure of cell $W_i(\mathbf{h})$ is denoted as $w_i(\mathbf{h})$:

$$w_i(\mathbf{h}) := \int_{W_i(\mathbf{h})} d\mu. \tag{18}$$

The dual to the power cell decomposition is the weighted Delaunay triangulation of the point set Y, denoted as $\mathcal{T}(\mathbf{h})$.

Theorem 2 (*Discrete Optimal Mass Transport*)**.** *For any given measures μ on X with convex support $\Omega \subset X$, and Dirac measure ν on Y, such that $\int_\Omega d\mu = \sum_{i=1}^k \nu_i$, $\nu_i > 0$, there must exist a height vector $\mathbf{h}$ unique up to adding a constant vector $(c, c, \ldots, c)$, the convex function in Eq. (16) induces the power cell decomposition of Ω as Eq. (17), such that the following* measure-preserving *constraints are satisfied for all cells,*

$$w_i(\mathbf{h}) = \nu_i, \ i = 1, 2, \ldots, k. \tag{19}$$

Furthermore, the gradient map $\nabla u_{\mathbf{h}}$ optimizes the following transportation cost

$$E(T) := \int_\Omega |x - T(x)|^2 d\mu(x), . \tag{20}$$

The existence and uniqueness have been first proven by Alexandrov [72] using a topological method. The existence has also been proven by Aurenhammer et al. [73].

Recently, Gu et al. [71] have given a novel proof for the existence and uniqueness based on variational principle. They proved the optimal height vector is the minimizer of the following convex energy:

$$E(\mathbf{h}) = \int_\Omega u_{\mathbf{h}}(x) d\mu(x) - \sum_{i=1}^{k} \nu_i h_i. \tag{21}$$

The gradient of the energy is given by:

$$\nabla E(\mathbf{h}) = (w_1(\mathbf{h}) - \nu_1, w_2(\mathbf{h}) - \nu_2, \ldots, w_k(\mathbf{h}) - \nu_k)^T. \tag{22}$$

The Hessian of the energy can be formulated as follows. Suppose two cells $W_i(\mathbf{h})$ and $W_j(\mathbf{h})$ intersect at a face $f_{ij}(\mathbf{h}) = W_i(\mathbf{h}) \cap W_j(\mathbf{h}) \cap \Omega$, then there is an edge in the weighted Delaunay triangulation connecting y_i and y_j, denoted as $e_{ij} \in \mathcal{T}(\mathbf{h})$. We define the *edge weight*

$$\tau_{ij} := \frac{1}{|y_j - y_i|} \int_{f_{ij}(\mathbf{h})} d\mu \tag{23}$$

The Hessian of $E(\mathbf{h})$ is the discrete Laplace–Beltrami matrix with the edge weight τ_{ij}, in detail:

$$\frac{\partial^2 E(\mathbf{h})}{\partial h_i \partial h_j} = \begin{cases} \sum_k \tau_{ik} & i = j \\ -\tau_{ij} & i \neq j, y_i \sim y_j \\ 0 & otherwise \end{cases} \tag{24}$$

Due to the convexity of the volume energy (Eq. (21)), the global minimum can be obtained efficiently using Newton's method.

3.3. Normal cycle

For the completeness, we briefly introduce the normal cycle theory. For a more thorough treatment, we refer readers to the work in [1].

3.3.1. Basic concepts

Intuitively, the normal cycle of a surface is its offset surface embedded in the higher dimensional Euclidean space. The concept of normal cycle of a smooth surface has been introduced in Section 1.2. Here we generalized the concept to polyhedral surfaces.

Suppose V is a convex body, whose boundary $M = \partial V$ is a polyhedral surface. The normal cone $NC_V(p)$ of a point $p \in V$ is the set of support vectors. The normal cycle of M is given by $N(M) := \{(p, \mathbf{n}) | p \in M, \mathbf{n} \in NC_V(p)\}$. For general volume V with a polyhedral boundary surface M, one can triangulate V by tetrahedra, $V = \bigcup_i t_i, i = 1, 2, \ldots, n$. The normal cycle of V and M is defined by inclusion–exclusion formula:

$$N(V) = N(M) := \sum_{k=1}^{n} (-1)^{k+1} \sum_{1 \leq i_1 < \cdots i_k \leq n} N(\cap_{j=1}^{k} t_{i_j}).$$

It can be shown that the normal cycle $N(V)$ is independent of triangulations. Similarly, the set-valued mapping from M to its normal cycle $N(M)$ is denoted as $i : M \to N(M), i(p) = (p, \mathbf{n}), \mathbf{n} \in NC_V(p)$.

3.3.2. Unified curvature

Normal cycles are embedded in the space $\mathbb{E}^3 \times \mathbb{E}^3$, with global coordinates $(x^1, x^2, x^3, y^1, y^2, y^3)$. The curvatures of smooth surface and discrete mesh can be unified using the following three differential 2-forms:

$$\begin{aligned} \omega^A &= y^1 dx^2 \wedge dx^3 + y^2 dx^3 \wedge dx^1 + y^3 dx^1 \wedge dx^2 \\ \omega^G &= y^1 dy^2 \wedge dy^3 + y^2 dy^3 \wedge dy^1 + y^3 dy^1 \wedge dy^2 \\ \omega^H &= y^1 (dx^2 \wedge dy^3 + dy^2 \wedge dx^3) + \\ &\quad y^2 (dx^3 \wedge dy^1 + dy^1 \wedge dx^3) + \\ &\quad y^3 (dx^1 \wedge dy^2 + dy^1 \wedge dx^2). \end{aligned} \tag{25}$$

Let $B \subset \mathbb{E}^3$ be a Borel set, the Gaussian curvature measure for a smooth surface S is

$$\phi_S^G(B) := \int_{B \cap S} G(p) dp,$$

where $G(p)$ is the Gaussian curvature of S at point p. Similarly, the mean curvature measure is given by

$$\phi_S^H(B) := \int_{B \cap S} H(p) dp,$$

where $H(p)$ is the measure curvature of S at point p. For discrete mesh, the Gaussian curvature measure is given by

$$\phi_M^G(B) := \sum_{v \in B \cap M} K(v), \tag{26}$$

where $K(v)$ is the discrete Gaussian curvature. The mean curvature measure is given by

$$\phi_M^H(B) := \sum_{e \in M} length(e \cap B) \beta(e),$$

where $\beta(e)$ is the angle between the normals to the faces adjacent to the edge e. The sign of $\beta(e)$ is chosen to be positive if e is convex and negative otherwise.

The curvature measure of a surface (no matter smooth or discrete) is equal to the integration of the specific differential form on its normal cycle,

$$\begin{aligned} \int_{N(M)} \omega^G_{|i(B \cap M)} &= \phi_M^G(B) \\ \int_{N(M)} \omega^H_{|i(B \cap M)} &= \phi_M^H(B) \\ \int_{N(M)} \omega^A_{|i(B \cap M)} &= Area(B \cap M) \end{aligned} \tag{27}$$

where M can be a smooth surface or a discrete mesh.

3.3.3. Curvature measure convergence

Given a Riemannian surface $(S, \mathbf{g})$, a conformal parameterization is denoted as $\varphi : S \to D$, where D is the parameter domain. Then we sample D and construct a triangulation T of D to produce a discrete mesh M, the piecewise linear mapping from M to T is denoted as $\tau : M \to T$. The composition map is denoted as $\pi = \varphi^{-1} \circ \tau : M \to S$. The following curvature measure convergence theorem was proven in [3] using normal cycle theory:

Theorem 3 (*Curvature Measure Convergence*)**.** *Let S be a surface embedded in $\mathbb{E}^3$. For any given upper bound ε of the circumradius, the Delaunay refinement algorithm on the conformal parameter domain will produce a triangulation with bounded aspect ratio, and induce a polyhedral triangular mesh M, which satisfies the following properties: let $B \subset \mathbb{E}^3$ be a relative interior of a union of triangles of M, then*

$$\begin{aligned} |\phi_M^G(B) - \phi_S^G(\pi(B))| &< K\varepsilon \\ |\phi_M^H(B) - \phi_S^H(\pi(B))| &< K\varepsilon \end{aligned} \tag{28}$$

where for fixed S

$$K = O(area(B \cap S)) + O(length(\partial B \cap S)).$$

This means, if one can construct triangulations with bounded minimal corner angle and the edge length, the resulting meshes have the curvature measure convergence property.

In our proposed method, given a topological disk $(S, \mathbf{g})$ embedded in $\mathbb{E}^3$, we compute the conformal parameterization $\varphi : S \to \mathbb{D}$, where $\mathbb{D}$ is the planar disk. Furthermore, we construct an area-preserving parameterization $\tau : NC(S) \to \mathbb{D}$. Then we uniformly sample on the image of τ, the sample set is denoted as P. On the image of φ, we start with P and use Delaunay refinement method (such as Chow's method) to construct a triangulation T with bounded corner angles, in turn construct a discrete M by T, then by the above theorem, M has the curvature measure estimation as in Eq. (28). Hence, we obtain the following corollary.

Corollary 1. *The curvature adaptive remeshing algorithm produces meshes with Gaussian curvature measure , mean curvature measure, area measure convergence property.*

4. Computational algorithm

In this section, we explain all the algorithms in detail.

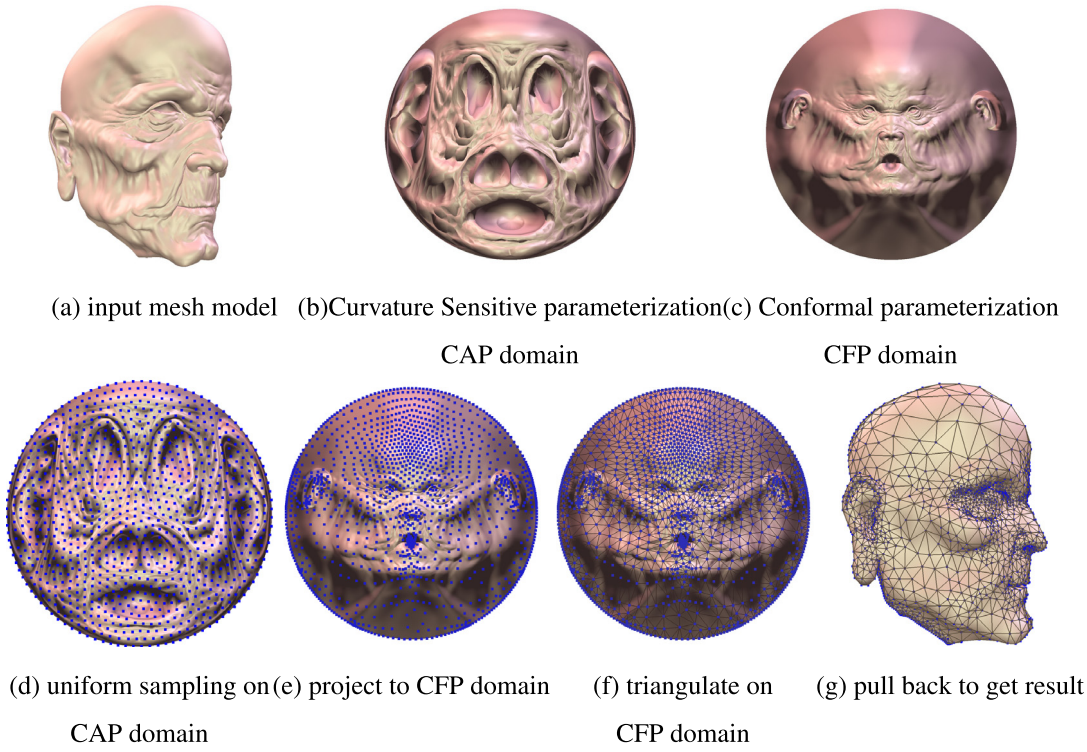


Fig. 3. Remeshing Algorithm Pipeline. Frame (a) shows the input mesh. Frame (b) and (c) show the CAP and CFP, respectively. The second row illustrates the detailed steps of our remeshing algorithm: first, we uniformly sample the CAP domain, as shown in Frame (d); second, we map the samples onto CFP domain, illustrated by Frame (e); third, we perform the Delaunay Triangulation algorithm on the CFP domain, shown in Frame (f); finally we pull back the triangulation to the original surface, to obtain the remeshing result, illustrated in Frame (g).

4.1. Pipeline

The algorithm pipeline is summarized in Alg. 1 and illustrated by Fig. 3. The input mesh is a genus zero mesh with a single boundary. The mesh quality of the input mesh could be very low, which does not affect our algorithm. The conformal parameterization (CFP) of the input surface uses the dynamic discrete surface Yamabe flow method [50], which is robust enough to handle low quality meshes. The area-preserving parameterization (APP) of the normal cycle is based on the discrete optimal mass transportation using variational approach [71], which has also been adopted in [74–76]. The curvature sensitive parameterization (CAP) is also based on the discrete optimal mass transportation method.

Algorithm 1: Remeshing Algorithm Pipeline.

Input: The input mesh M and the number of samples n
Output: The remeshing result $\tilde{M}$ with n vertices
1. Compute the Conformal parameterization of the input surface (CFP);
2. Compute the Curvature Adaptive Parameterization of the normal cycle (CAP);
3. Sample n points $\mathbf{P}$ uniformly on CAP domain;
4. Map $\mathbf{P}$ to CFP domain to get $\mathbf{Q}$;
5. Delaunay triangulate $\mathbf{Q}$ to get a triangulation T;
6. Pull back T to the original mesh M to get the final result.

In the following, we explain the details of each step. The theoretic proofs for the existence of solutions to dynamic discrete Yamabe flow can be found in [50]. The theoretic aspects of the discrete optimal transportation map are covered by [71].

4.2. Dynamic discrete surface Yamabe flow

The dynamic Yamabe flow algorithm mainly optimizes the following convex energy in Eq. (11) with the linear constraint $\sum_i u_i = 0$. The gradient is the difference between the target curvature and the current curvature, as shown in Eq. (12). The Hessian matrix is the classical discrete Laplace–Beltrami matrix in Eq. (13), consisting of the cotangent edge weight in Eq. (14). During the flow, the triangulation is preserved to be Delaunay by edge swapping operator. The details of the algorithm can be found in Alg. 2.

The dynamic Yamabe flow can handle meshes with low qualities. In our current work, we set the target curvature of the interior vertices to be zero everywhere, the target curvatures of the boundary vertices to be constant. After obtaining the target edge length, we can flatten the whole mesh face by face, such that the input simply connected mesh is mapped onto a planar convex domain.

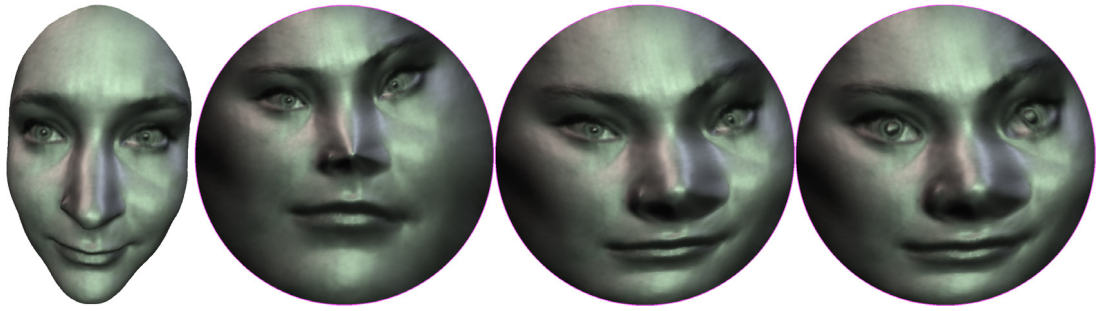
4.3. Optimal mass transportation map

In the current work, the source domain Ω is the canonical convex domain in $\mathbb{R}^2$, the target is a set of discrete points $Y = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\}$ which densely samples Ω. The source measure on Ω is the uniform measure $\mu = 1$ everywhere. The target measure on Y is prescribed by the user, $\nu = \{\nu_1, \nu_2, \ldots, \nu_k\}$, such that $\sum_{i=1}^{k} \nu_i$ equals the total area of Ω.

4.3.1. The target measure

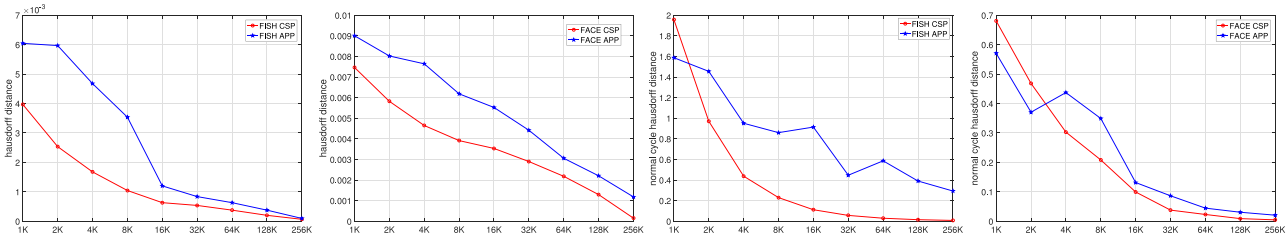
In order to compute the area-preserving parameterization of the normal cycle, we need to set the area element as the target measure ν.

Given the input mesh $M = (V, E, F)$, for each face $f \in F$, we compute the normal to it $\mathbf{n}_f$. The vertex normal is defined



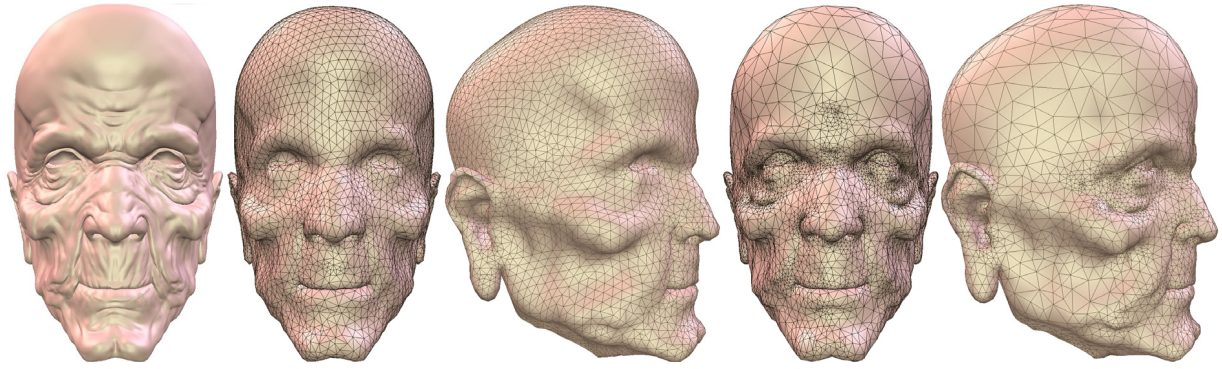
(a) Sophie model (b) CFP (c) APP (d) CAP

Fig. 4. Different parameterization methods for the Sophie model: CFP, CAP and APP (Area Preserving Parameterization).



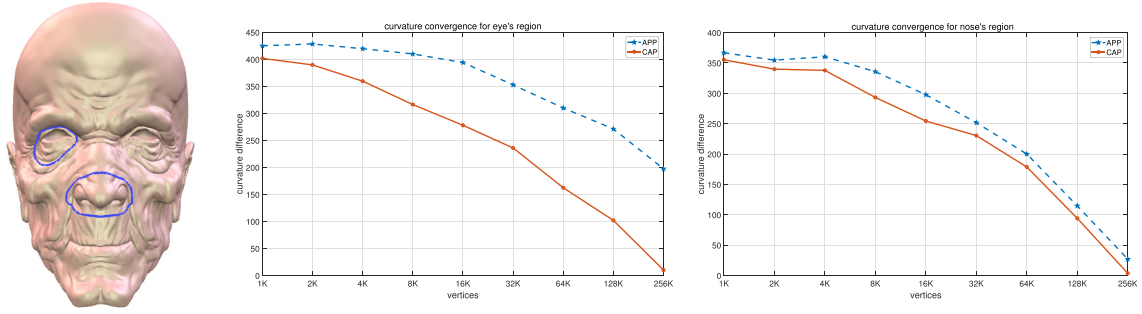
(a) hausdorff distance fish model (b) hausdorff distance face model (c) normal cycle distance fish model (d) normal cycle distance face model

Fig. 5. Comparison of Hausdorff distances between the surfaces and the normal cycles based on APP and CAP methods.



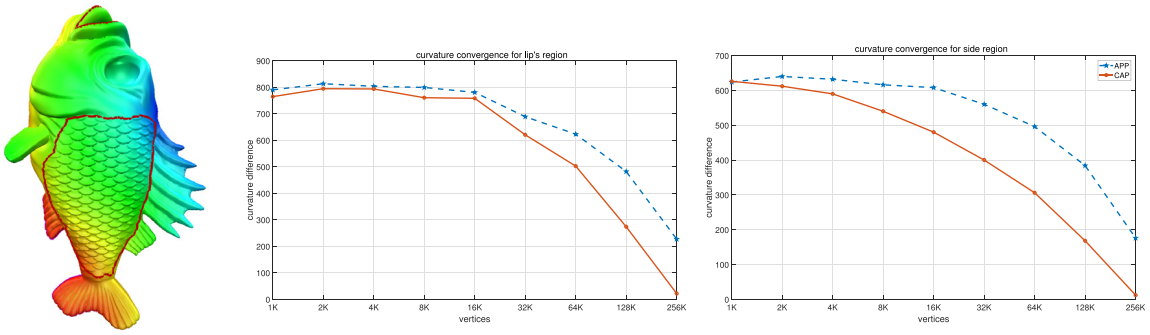
(a) (b) (c) (d) (e)

Fig. 6. Comparison between remeshing based on curvature adaptive sampling and uniform sampling. (a) is the original oldman face model with 140k vertices. (b) and (c) are the remeshing results by APP with 5k vertices. (d) and (e) are obtained by CAP with 5k vertices. The curvature adaptive remeshing preserves the geometric details better.



(a) (b) APP remeshing (c) CAP remeshing

Fig. 7. Comparison of the curvature convergence of the remeshing results based on APP and CAP. (a) shows the original oldman face model with two ROIs, the eye area and the nose region. (b) and (c) show the curvature convergence curves, the blue and red curves are based on APP and CAP respectively. The horizontal axis indicates the number of vertices, the vertical axis is the curvature error. It is obvious the CAP remeshing has higher curvature convergence rate.



(a) (b) (c)

Fig. 8. Comparison of the curvature convergence of the fish model. (a) is the fish model with 2 ROIs, the lip and the side regions. (b) and (c) show the curvature convergence curves for the lip and side regions respectively. The blue (red) curves show the results based on APP (CAP). It is obvious that the CAP remeshing has faster convergence rate.

Algorithm 2: Dynamic Discrete Surface Yamabe Flow.

Input: The input mesh M and the target curvature $\bar{K}$, threshold ε
Output: The edge length which realizes the target curvature
Compute the initial edge lengths $\{\beta_{ij}\}$;
Initialize the conformal factor to be zeros;
while *true* **do**
  Compute the edge lengths using Eq. (9);
  Update the triangulation to be Delaunay by edge swapping;
  Compute the corner angles using Eq. (6);
  Compute the cotangent edge weights using Eq. (14);
  Compute the vertex curvature using Eq. (7);
  if $\forall |K_i - K_i(\mathbf{h})| < \varepsilon$ **then**
    Break;
  Compute the gradient of the Yamabe energy using Eq. (12);
  Compute the Hessian of the Yamabe energy using Eq. (13);
  Solve the linear system $Hess(\mathbf{u})\delta\mathbf{u} = \nabla E(\mathbf{u})$
  $\mathbf{u} \leftarrow \mathbf{u} + \delta\mathbf{u}$;
return *the edge length* $\{l_{ij}\}$

as a linear combination of the normals to the surrounding faces, weighted by the areas,

$$\mathbf{n}_v = \frac{\sum_{v\in f} \mathbf{n}_f A_f}{|\sum_{v\in f} \mathbf{n}_f A_f|},$$

where A_f is the area of the face f. The discrete Riemannian metric of the normal cycle $N(M)$ is denoted as $d : E \to \mathbb{R}_+$, for each edge $[v_i, v_j]$,

$$d_{ij} = \sqrt{|v_j - v_i|^2 + |\mathbf{n}_j - \mathbf{n}_i|^2},$$

where $\mathbf{n}_i$ and $\mathbf{n}_j$ are the normals to the vertices v_i and v_j respectively.

For each vertex $v_i \in V$, the target point $\mathbf{q}_i$ is defined as its planar conformal parameter, the target measure ν_i is defined as

$$\nu_i := \frac{1}{3}\sum_{v_i\in f} A_f + \lambda|K(v_i)|, \tag{29}$$

namely, a linear combination of the surface area element and the Gaussian sphere area element, where λ is a positive constant.

4.3.2. Power diagram and weighted delaunay triangulation

For each target point $\mathbf{q}_i \in Y$, we construct a hyperplane in $\mathbb{R}^3$, $\pi_i(\mathbf{h}) : \langle \mathbf{q}_i, \mathbf{p}\rangle + h_i$, $i = 1, 2, \ldots, k$. Then we compute the *upper envelope* of these hyperplanes. For each hyperplane $\pi_i(\mathbf{h})$, we construct a dual point $\pi_i^*(\mathbf{h}) \in \mathbb{R}^3$ as follows: assume the coordinates of $\mathbf{q}_i \in \mathbb{R}^2$ are (x_i, y_i), then the dual point is $\pi_i^*(\mathbf{h}) = (x_i, y_i, -h_i)$, $i = 1, 2, \ldots, k$. Then we compute the *convex hull* of $\{\pi_1^*(\mathbf{h}), \pi_2^*(\mathbf{h}), \ldots, \pi_k^*(\mathbf{h})\}$ using incremental convex hull algorithm as described in [77], denoted as $\mathcal{C}(\mathbf{h})$. We remove all the faces of $\mathcal{C}(\mathbf{h})$, facing upwards. The dual of $\mathcal{C}(\mathbf{h})$ is the upper envelope of the hyperplanes, denoted as $\mathcal{E}(\mathbf{h})$. The projection of the upper envelope is the *power diagram*, denoted as $\mathcal{V}(\mathbf{h})$. The projection of the convex hull is the *weighted Delaunay triangulation* of Y, denoted as $\mathcal{T}(\mathbf{h})$.

Algorithm 3: Discrete Optimal Mass Transportation Map

Input: A convex domain $\Omega \subset \mathbb{R}^2$ and a set of discrete points $Y = \{\mathbf{q}_1, \cdots, \mathbf{q}_n\}$, discrete target measure $\nu = \{\nu_1, \cdots, \nu_n\}$, such that $\sum_i \nu_i = Area(\Omega)$
Output: A partition of Ω, $\Omega = \cup_i W_i$, such that $W_i \mapsto \mathbf{q}_i$ is the optimal mass transportation map.
Translate and scale Y, such that $Y \subset \Omega$
Initialize the height vector $\mathbf{h}$, such that $h_i \leftarrow 1/2\langle \mathbf{q}_i, \mathbf{q}_i\rangle$
while *true* **do**
  for $i \leftarrow 1$ ***to*** k **do**
    Construct the plane $\pi_i(\mathbf{h}) : \langle \mathbf{q}_i, \mathbf{p}\rangle + h_i$
    Compute the dual point of the plane $\pi_i^*(\mathbf{h})$
  Construct the convex hull $\mathcal{C}(\mathbf{h})$ of the dual points $\{\pi_i^*(\mathbf{h})\}$
  Compute the dual of the convex hull to obtain the upper envelope $\mathcal{E}(\mathbf{h})$ of the planes $\{\pi_i(\mathbf{h})\}$
  Project $\mathcal{C}(\mathbf{h})$ to obtain the weighted Delaunay triangulation $\mathcal{T}(\mathbf{h})$ of Y
  Project $\mathcal{E}(\mathbf{h})$ to obtain the power cell decomposition $\mathcal{V}(\mathbf{h})$ of Ω
  for $i \leftarrow 1$ ***to*** k **do**
    Compute the area of $W_i(\mathbf{h})$, denoted as $w_i(\mathbf{h})$
  Construct the gradient Eq. (22);
  Construct the Hessian matrix Eq. (23) and Eq. (24);
  Solve the linear equation $Hess(\mathbf{h})\delta\mathbf{h} = \nabla E(\mathbf{h})$
  $\lambda \leftarrow 1$
  Compute the power diagram $\mathcal{A}(\mathbf{h} + \lambda\delta\mathbf{h})$ of Ω
  while $\exists w_i(\mathbf{h} + \lambda(\delta\mathbf{h}))$ *is empty* **do**
    $\lambda \leftarrow 1/2\lambda$
    Compute the power diagram $\mathcal{A}(\mathbf{h} + \lambda\delta\mathbf{h})$ of Ω
  $\mathbf{h} \leftarrow \mathbf{h} + \lambda\delta\mathbf{h}$
  if $\forall |w_i(\mathbf{h}) - \nu_i| < \varepsilon$ **then**
    Break
return *the mapping* $\{W_i(\mathbf{h}) \mapsto \mathbf{q}_i, i = 1, 2, \cdots, k\}$

4.3.3. Convex optimization

The main focus of the algorithm is to optimize the convex energy Eq. (21) with the linear constraint $\sum_{i=1}^{k} h_i = 0$. The gradient is the difference between the target measure and the current measure of each power cell in Eq. (22). Each Delaunay edge is dual to a unique Voronoi edge, the ratio between lengths is defined as the edge weight in Eq. (23). The Hessian matrix is the discrete Laplace–Beltrami operator in Eq. (24). We use Newton's method to iteratively optimize the energy. During the optimization process, we need to ensure the height vector $\mathbf{h}$ is admissible, namely, in the power cell decomposition $\mathcal{V}(\mathbf{h})$, each cell $W_i(\mathbf{h})$ is non-empty. In our algorithm, if the height vector exceeds the admissible space, the step length is reduced by half. The details of the algorithm can be found in Alg. 3.

4.4. Remeshing

The conformal parameterization (CFP) is denoted as $\varphi : M \to \mathbb{D}_0$, the area-preserving parameterization of the normal cycle (CAP) is denoted as $\psi : \mathbb{D}_0 \to \mathbb{D}_1$, where both $\mathbb{D}_0$ and $\mathbb{D}_1$ are the planar unit disks, with different measures. We uniformly sample the CAP domain $\mathbb{D}_1$ to obtain the sample point set P. The inverse of the CAP ψ^{-1} maps P to the CFP domain $\mathbb{D}_0$, furthermore the inverse of the CFP φ^{-1} maps the sample points to the original mesh M. On the CFP domain, we compute the restricted Delaunay triangulation $\mathcal{T}$, then φ^{-1} pulls back $\mathcal{T}$ to the mesh M, then induce a triangulation on M. Because φ is conformal, $\mathcal{T}$ is Delaunay on $\mathbb{D}_0$, it is also Delaunay on M. In this way, we obtain the remeshing result $\tilde{M}$.

5. Experimental results

We implemented the proposed algorithms in generic C++, using VISUAL Studio on Windows platform. All the experiments are carried out on a laptop with 2.3 GHz dual core CPU and 8GB memory. We report our results in the following four subsections, which demonstrate that our algorithm allows users to control the sampling distribution and produce high quality meshes. We have conducted our experiments on surfaces either manually constructed (the old man head) or scanned from real objects: the human face model 4, the fish model 8 and some other models.

Normal cycle hausdorff distance. Fig. 4 shows different parameterization methods for the Sophie facial surface model. Fig. 5 shows the comparison of Hausdorff distances between the surfaces and the normal cycles based on APP and CAP methods. The horizontal axis indicates the number of samples, the vertical axis shows the Hausdorff distance. The blue (red) curve shows the distance between the original surface and the remeshing results based on APP (CAP). It is obvious that curvature adaptive sampling method achieves smaller Hausdorff distances between the normal cycles than uniform sampling method.

Curvature convergence. Fig. 6 shows the reconstruction results based on uniform sampling (APP) and curvature adaptive parameterization (CAP). By visual comparison, it is clear that the CAP method better preserves subtle geometric features, such as the eye lids, the wrinkles of the skin and the contour of the ears.

Furthermore, Fig. 7 shows the comparison between the curvature convergence rates quantitatively. In frame (a), a neighborhood of the right eye and a neighborhood of the nose tip are selected. The discrete Gaussian curvature measure of each region is calculated using Eq. (26). Frame (b) shows the total curvature errors in the eye region on the original surface and the reconstructed mesh with different sampling rate based on APP. Frame (c) shows the curvature error-sampling rate curve based on CAP. It is clear that the CAP based method outperforms the APP based one. Fig. 8 illustrates the results for the same experiment for the fish model, and demonstrates that the curvature adaptive remeshing achieves higher curvature convergence rate.

Table 1
Comparison with state-of-the-art methods.

Input	#Samples	Methods	Ndist	Cdist	Time (s)
Oldman (140k)	20k	[RAR]	0.105	0.039	**3.87**
		[Vorpaline]	0.107	0.029	16.984
		[CVT]	0.098	0.038	6.484
		[APP]	**0.055**	0.024	6.687
		[CAP]	0.067	**0.015**	6.203
Buddha (47k)	30k	[MAI]	0.146	0.085	555
		[RAR]	0.168	0.109	**3.04**
		[Vorpaline]	0.142	0.102	19.203
		[CVT]	0.112	0.085	5.141
		[APP]	0.103	0.050	4.86
		[CAP]	**0.099**	**0.023**	5.532
Skull (19k)	15k	[MAI]	0.136	0.085	219
		[RAR]	0.076	0.042	**1.09**
		[Vorpaline]	0.060	0.025	7.953
		[CVT]	0.058	0.022	2.344
		[APP]	**0.045**	0.019	2.281
		[CAP]	0.049	**0.011**	1.97
Bunny (23k)	19k	[MAI]	0.055	0.094	245
		[RAR]	0.045	0.089	**1.37**
		[Vorpaline]	0.048	0.091	7.953
		[CVT]	0.061	0.030	3.047
		[APP]	0.047	0.016	2.765
		[CAP]	**0.035**	**0.009**	2.891
Bimba (15k)	12k	[MAI]	0.085	0.057	255
		[RAR]	0.092	0.029	**0.97**
		[Vorpaline]	0.084	0.023	5.766
		[CVT]	0.077	0.022	2.188
		[APP]	**0.061**	0.021	1.765
		[CAP]	0.075	**0.013**	1.953

Comparison with state-of-the-art methods. We compare our curvature adaptive remeshing method with several state-of-the-art approaches, as explained in the following: Hu et al. proposed a feature preserving surface remeshing algorithm with bounded err in [78], denoted as MAI; Botsch et al. proposed an adaptive remeshing method for real time mesh deformation in [79], denoted as RAR; Levy and Bonneel proposed a variational anisotropic surface meshing algorithm with Voronoi parallel linear enumeration in [80], denoted as Vorpaline; Yan et al. proposed isotropic remeshing algorithm with fast and exact computation of restricted Voronoi diagram in [18], denoted as CVT.

Figs. 9 and 10 visually compare the resulting results obtained by the state-of-the-art methods and our proposed methods. Quantitative comparison is summarized in Table 1, where we compute the mean value of the normal distance (Ndist) and the curvature distance (Cdist). First, each vertex on the remeshed result is pulled back to the original surface, then the deviation between the normals on the pair of corresponding points is calculated. If the source point is interior of a triangle face, we use barycentric coordinates to linearly interpolate the normals at the vertices of the face. The curvature distance is carried out similarly. From Table 1, it is obvious that our proposed algorithm gives the optimal results in terms of normal and curvature distances.

6. Conclusions

This work proposes a novel framework for curvature adaptive remeshing based on normal cycle theory, which guarantees the curvature measures of the resulting meshes converge to those of the smooth surface. Furthermore, the normal cycle theory guides the sampling scheme. The method uniformly samples the normal cycle, instead of the original surface, this improves the speed of curvature convergence. Our experimental results demonstrate the efficiency and efficacy of the proposed method.

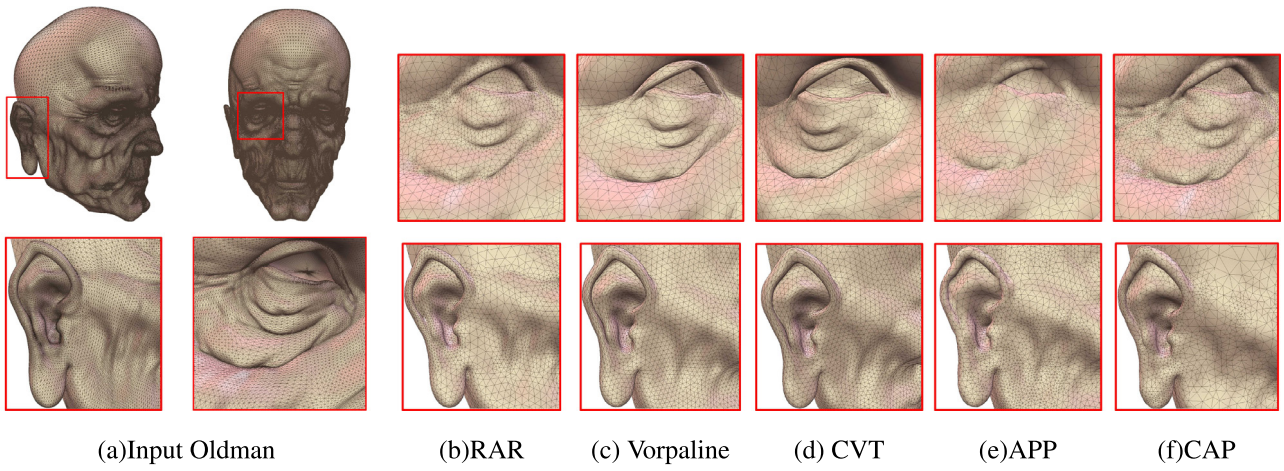


Fig. 9. Comparison with state-of-the-art approaches for the oldman model.

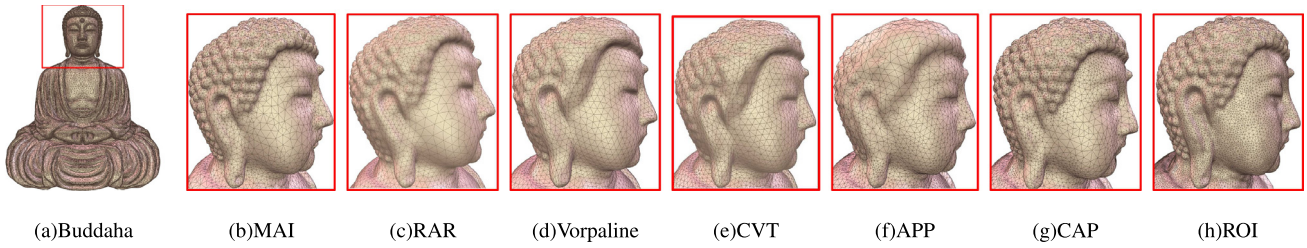


Fig. 10. Comparison with state-of-the-art approaches for the Buddha model, ROI means region of interest, which is select manually.

In the future, we will generalize the proposed isotropic remeshing method to anisotropic method for better adapting to the geometry of the input surfaces; furthermore, we will combine normal cycle theory with Centroidal Voronoi Tessellation (CVT) method, design meshing algorithm with better curvature convergence.

Acknowledgments

The project is partially supported by NSFC 61772379, 61772105, 61432003 and 61720106005, NSF CMMI-1762287 Collaborative Research: Computational Framework for Designing Conformal Stretchable Electronics, Ford URP Topology Optimization of Cellular Mesostructure's Nonlinear Behaviors for Crash Safety and NSF DMS-1737812 Collaborative Research: ATD: Theory and Algorithms for Discrete Curvatures on Network Data from Human Mobility and Monitoring.

References

[1] Morvan JM. Generalized curvatures. Geometry and computing, Springer Verlag; 2008.
[2] Cohen-Steiner D, Morvan J-M. Restricted Delaunay triangulations and normal cycle. In: Proceedings of the nineteenth annual symposium on computational geometry. SCG '03, 2003, p. 312–21.
[3] Li H, Zeng W, Morvan JM, Chen L, Gu XD. Surface meshing with curvature convergence. IEEE Trans Vis Comput Graphics 2014;20(6):919–34. http://dx.doi.org/10.1109/TVCG.2013.253.
[4] Amenta N, Bern M, Kamvysselis M. A new Voronoi-based surface reconstruction algorithm. In: Proceedings of the 25th annual conference on computer graphics and interactive techniques. SIGGRAPH '98, New York, NY, USA: ACM; 1998, p. 415–21. http://dx.doi.org/10.1145/280814.280947, URL http://doi.acm.org/101145/280814280947.
[5] Alliez P, Meyer M, Desbrun M. Interactive geometry remeshing. ACM Trans Graph 2002;21(3):347–54.
[6] Alliez P, Attene M, Gotsman C, Ucelli G. Recent advances in remeshing of surfaces. In: Shape analysis and structuring. Springer; 2008, p. 53–82.
[7] Surazhsky V, Gotsman C. Explicit surface remeshing. In: Proceedings of eurographics symposium on geometry processing. Aachen, Germany; 2003, p. 17–28.
[8] Frey PJ, Borouchaki H. Geometric surface mesh optimization. Comput Vis Sci 1998;1(3):113–21.
[9] Edelsbrunner H, Shah NR. Triangulating topological spaces. Int J Comput Geom Appl 1997;7(4):365–78.
[10] Cheng S-W, Dey TK, Levine JA. A practical Delaunay meshing algorithm for a large class of domains. In: Proc. 16th international meshing roundtable. Sandia National Laboratories; 2007, p. 477–94.
[11] Amenta N, Bern MW. Surface reconstruction by Voronoi filtering. In: Proceedings of the fourteenth annual symposium on computational geometry, Minneapolis, Minnesota, USA, June 7-10, 1998. 1998, p. 39–48.
[12] Morvan J-M, Thibert B. On the approximation of the area of a surface. In: Tech. Rep. RR-4375. INRIA; 2002.
[13] Boissonnat JD, Oudot S. Provably good sampling and meshing of surfaces. Graph Models 2005;67:405–51.
[14] Du Q, Faber V, Gunzburger M. Centroidal Voronoi tessellations: Applications and algorithms. SIAM Rev 1999;41(4):637–76.
[15] Alliez P, d. Verdière EC, Devillers O, Isenburg M. Isotropic surface remeshing. In: Proceedings of the shape modeling international 2003. Washington, DC: IEEE Computer Society; 2003, p. 49.
[16] Lévy B, Liu Y. L_p centroidal voronoi tesselation and its applications. In: ACM transactions on graphics (SIGGRAPH conference proceedings). 2010.
[17] Chen Z, Cao J, Wang W. Isotropic surface remeshing using constrained centroidal Delaunay mesh. Comput Graph Forum 2012;31(7–1):2077–85.
[18] Yan D, Lévy B, Liu Y, Sun F, Wang W. Isotropic remeshing with fast and exact computation of restricted voronoi diagram. Comput Graph Forum 2009;28(5):1445–54.
[19] Alliez P, Cohen-Steiner D, Devillers O, Lévy B, Desbrun M. Anisotropic polygonal remeshing. In: ACM SIGGRAPH 2003 papers. New York, NY: ACM; 2003, p. 485–93.
[20] Jiao X, Colombi A, Ni X, Hart JC. Anisotropic mesh adaptation for evolving triangulated surfaces. In: Proc. 16th international meshing roundtable. 2006.
[21] Boissonnat JD, Wormser C, Yvinec M. Locally uniform anisotropic meshing. In: Proc. 24th ann. symp. on comput. geom.. 2008.
[22] Canas GD, Gortler SJ. Surface remeshing in arbitrary codimensions. Vis Comput 2006;22(9):885–95.
[23] Lai Y-K, Zhou Q-Y, Hu S-M, Wallner J, Pottmann H. Robust feature classification and editing. IEEE Trans Vis Comput Graphics 2007;13(1):34–45.

[24] Lévy B, Bonneel N. Variational anisotropic surface meshing with voronoi parallel linear enumeration. In: Proc. 21st international meshing roundtable. 2012, p. 349–66.
[25] Dassi F, Si H. A curvature-adapted anisotropic surface re-meshing method. Springer International Publishing; 2015, p. 19–41.
[26] Chew LP. Guaranteed-quality triangular meshes. Department of Computer Science Tech Report 89-983, Cornell University; 1989.
[27] Chew LP. Guaranteed-quality mesh generation for curved surfaces. In: Proc. 9th ann. sympos. computat. geom.. 1993, p. 274–80.
[28] Ruppert J. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. J Algorithms 1995;18:548–85.
[29] Shewchuk JR. Delaunay refinement algorithms for triangular mesh generation. Comput Geom 2002;22(1–3):21–74.
[30] Cheng S-W, Dey TK, Ramos EA, Ray T. Sampling and meshing a surface with guaranteed topology and geometry. SIAM J Comput 2007;37(4):1199–227. http://dx.doi.org/10.1137/060665889, URL http://dx.doi.org/10.1137/060665889, http://arxiv.org/abs/http://dx.doi.org/10.1137/060665889.
[31] Dey TK, Ray T. Polygonal surface remeshing with Delaunay refinement. Eng Comput 2010;26(3):289–301. http://dx.doi.org/10.1007/s00366-009-0162-1, URL http://dx.doi.org/10.1007/s00366-009-0162-1.
[32] Dey TK, Levine JA. Delaunay meshing of isosurfaces. In: Proc. shape modeling international. 2007, p. 241–50.
[33] Cheng S-W, Dey TK, Shewchuk JR. Delaunay mesh generation. CRC Press; 2012.
[34] Thurston W. The geometry and topology of 3-manifolds. Princeton University Press; 1997.
[35] Rodin B, Sullivan D. The convergence of circle packings to the Riemann mapping. J. Differ Geom 1987;26(2):349–60, http://projecteuclid.org/euclid.jdg/1214441375.
[36] He Z-X, Schramm O. On the convergence of circle packings to the Riemann map. Invent Math 1996;125(2):285–305. http://dx.doi.org/10.1007/s002220050076, URL http://dx.doi.org/10.1007/s002220050076.
[37] Chow B, Luo F. Combinatorial Ricci flows on surfaces. J. Differ Geom 2003;63(1):97–129.
[38] Gu DX, Zeng W, Luo F, Yau S-T. Numerical computation of surface conformal mappings. Comput Methods Funct Theory 2011;11(2):747–87.
[39] de Verdiére YC. Un principe variationnel pour les empilements de cercles. Invent Math 1991;104:655–69.
[40] Stephenson K. Introduction to circle packing: The theory of discrete analytic functions. Cambridge University Press; 2005.
[41] Bowers PL, Stephenson K. Uniformizing dessins and Belyi maps via circle packing. Mem Amer Math Soc 2004;170(805).
[42] Guo R. Local rigidity of inversive distance circle packing. Trans Amer Math Soc 2011;363:4757–76.
[43] Luo F. Rigidity of polyhedral surfaces, III. Geom Topol 2011;15(4):2299–319.
[44] Luo F. Combinatorial yamabe flow on surfaces. Contemp. Math. 2004;6(5):765–80.
[45] Springborn B, Schröder P, Pinkall U. Conformal equivalence of triangle meshes. ACM Trans Graph 2008;27(3):77:1–77:11. http://dx.doi.org/10.1145/1360612.1360676, URL http://doi.acm.org/10.1145/1360612.1360676.
[46] Glickenstein D. A combinatorial yamabe flow in three dimensions. Topology 2005;44(4):791–808.
[47] Glickenstein D. A maximum principle for combinatorial Yamabe flow. Topology 2005;44(4):809–25.
[48] Alexander Bobenko BS. Discrete conformal maps and ideal hyperbolic polyhedra. Geom Topol 2015;19:2055–215.
[49] GUO R. Combinatorial yamabe flow on hyperbolic surfaces with boundarY. Commun Contemp Math 2011;13(05):827–42. http://dx.doi.org/10.1142/S0219199711004464, http://arxiv.org/abs/http://www.worldscientific.com/doi/pdf/10.1142/S0219199711004464, URL http://www.worldscientific.com/doi/abs/10.1142/S0219199711004464.
[50] Gu X, Luo F, Sun J, Wu T. A discrete uniformization theorem for polyhedral surfaces. J Differ Geom 2018;109(2):223–56.
[51] Gu X, Luo F, Sun J, Wu T. A discrete uniformization theorem for polyhedral surfaces. J Differ Geom 2018;109(3):431–66.
[52] Zeng W, Gu X. Ricci flow for shape analysis and surface registration. Springer briefs in mathematics, Springer New York; 2013.
[53] Zhang M, Guo R, Zeng W, Luo F, Yau S-T, Gu X. The unified discrete surface Ricci flow. Graph. Models 2014;76(5):321–39. http://dx.doi.org/10.1016/j.gmod.2014.04.008, URL http://dx.doi.org/10.1016/j.gmod.2014.04.008.
[54] Glickenstein D. Problems in combinatorial and numerical ricci flow, Talk in Workshop: Perspective Of The Ricci Flow (2 2013).
[55] Glickenstein D. Discrete conformal variations and scalar curvature on piecewise flat two and three dimensional manifolds. J Differential Geom 2011;87(2):201–38.
[56] Brägger W. Kreispackungen und Triangulierugen. Enseign Math 1992;38:201–17.
[57] Rivin I. Euclidean structures of simplicial surfaces and hyperbolic volume. Ann Math 1994;139:553–80.
[58] Leibon G. Characterizing the Delaunay decompositions of compact hyperbolic surface. Geom Topol 2002;6:361–91.
[59] Bobenko AI, Springborn BA. Variational principles for circle patterns and Koebe's theorem. Trans Amer Math Soc 2004;356(2):659–89.
[60] Guo R, Luo F. Rigidity of polyhedral surface II. Geom Topol 2009;13:1265–312.
[61] Springborn B. A variational principle for weighted Delaunay triangulation and hyperideal polyhedra. J Differential Geom 2008;78(2):333–67.
[62] Luo F. Rigidity of polyhedral surfaces. J Differential Geom 2014;96(1) 241–302.
[63] Luo F, Gu X, Dai J. Variational principles for discrete surfaces. Advanced lectures in mathematics, High Education Press and International Press; 2007.
[64] Hersonsky S. Boundary value problems on planar graphs and flat surfaces with integer cone singularities, I: The Dirichlet problem. J Reine Angew Math 2012;(670):65–92.
[65] Hersonsky S. The triple intersection property, three dimensional extremal length, and tiling of a topological cube. Topology Appl 2012;159(10–11):2795–805.
[66] Hersonsky S. Boundary value problems on planar graphs and flat surfaces with integer cone singularities, II: the mixed Dirichlet-Neumann problem. Differential Geom Appl 2011;29(3):329–47.
[67] Hersonsky S. Energy and length in a topological planar quadrilateral. European J Combin 2011;29(1):208–17.
[68] Morvan J-M, Thibert B. Approximation of the normal vector field and the area of a smoothsurface. Discrete Comput Geom 2004;32(3):383–400. http://dx.doi.org/10.1007/s00454-004-1096-4, http://dx.doi.org/10.1007/s00454-004-1096-4.
[69] Xu G. Convergence analysis of a discretization scheme for Gaussian curvature over triangular surfaces. Comput Aided Geom Design 2006;23(2):193–207. http://dx.doi.org/10.1016/j.cagd.2005.07.002, URLhttp://dx.doi.org/10.1016/j.cagd.2005.07.002.
[70] Hildebrandt K, Polthier K, Wardetzky M. On the convergence of metric and geometric properties of polyhedral surfaces. Geom Dedicata 2006;123(1):89–112. http://dx.doi.org/10.1007/s10711-006-9109-5, URL http://dx.doi.org/101007/s10711-006-9109-5.
[71] Gu X, Luo F, Sun J, Yau S-T. Variational principles for Minkowski type problems, discrete optimal transport, and discrete Monge-Ampere equations, arXiv:13025472 (2013) 1–13.
[72] Andreev EM. Convex polyhedra of finite volume in Lobachevsky space. (Russian) Mat Sb (N.S.) 1970;83(125):256–60.
[73] Aurenhammer. Power diagrams: properties, algorithms and applications. Siam J Comput. 2006;16(1):78–96.
[74] Su K, Cui L, Qian K, Lei N, Zhang J, Zhang M, Gu XD. Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation. Comput Aided Geom Design 2016;46(C):76–91.
[75] Su K, Chen W, Lei N, Cui L, Jiang J, Gu XD. Measure controllable volumetric mesh parameterization. Comput Aided Des 2016;78(C):188–98.
[76] Su K, Chen W, Lei N, Zhang J, Qian K, Gu X. Volume preserving mesh parameterization based on optimal mass transportation. Comput Aided Des 2017;82:42–56.
[77] Mark de Berge MvK, Otfried Cheong, Overmars M. Computational geometry: Algorithm and application. 3rd ed.. Springer Verlag; 2008.
[78] Hu K, Yan D-M, Bommes D, Alliez P, Benes B. Error-bounded and feature preserving surface remeshing with minimal angle improvement, IEEE Transactions on Visualization and Computer Graphics.
[79] Dunyach M, Vanderhaeghe D, Barthe L, Botsch M. Adaptive remeshing for real-time mesh deformation. In: Eurographics (Short Papers). 2013 p. 29–32.
[80] Lévy B, Bonneel N. Variational anisotropic surface meshing with voronoi parallel linear enumeration. In: Proceedings of the 21st international meshing roundtable. Springer; 2013, p. 349–66.