Computer Aided Geometric Design 63 (2018) 96–108

Contents lists available at ScienceDirect

Computer Aided Geometric Design

www.elsevier.com/locate/cagd

Conformal mesh parameterization using discrete Calabi flow

Hui Zhao [a,b], Xuan Li [e], Huabin Ge [c], Na Lei [d,h,*], Min Zhang [f,g], Xiaoling Wang [i], Xianfeng Gu [e]

[a] Tsinghua University, China
[b] Beijing Advanced Innovation Center for Imaging Technology, Capital Normal University, China
[c] Department of Mathematics, Beijing Jiaotong University, China
[d] DUT-RU International School of Information and Software Engineering, Dalian University of Technology, China
[e] Computer Science Department, State University of New York at Stony Brook, USA
[f] Brigham and Women's Hospital, Harvard Medical School, USA
[g] Center of Mathematical Sciences and Applications, Harvard University, USA
[h] Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, China
[i] School of Mechanical Engineering, University of Science and Technology Beijing, China

ARTICLE INFO

Article history:
Available online 2 May 2018

Keywords:
Calabi flow
Mesh parameterization
Conformal mapping

ABSTRACT

In this paper, we introduce discrete Calabi flow to the graphics research community and present a novel conformal mesh parameterization algorithm. Calabi energy has a succinct and explicit format. Its corresponding flow is conformal and convergent under certain conditions. Our method is based on the Calabi energy and Calabi flow with solid theoretical and mathematical base. We demonstrate our approach on dozens of models and compare it with other related flow based methods, such as the well-known Ricci flow and conformal equivalence of triangle meshes (CETM). Our experiments show that the performance of our algorithm is comparably the same with other methods. The discrete Calabi flow in our method provides another perspective on conformal flow and conformal parameterization.

1. Introduction

In this paper, we present a novel conformal flow-based method for conformal mapping. Our method is based on discrete Calabi energy and Calabi flow (Chen and He, 2008; Ge, 2012, 2018; Ge and Xu, 2016). Discrete Calabi flow is inspired by discrete Ricci flow (Chow et al., 2003; Luo, 2004; Jin et al., 2007a; Zhang et al., 2014), it is also a conformal flow which preserves the angles. Conformal parameterization can keep the shape of the original mesh and is especially useful in all kinds of applications.

Mesh mapping and parameterization are crucial operations in computer graphics modeling. Researchers have designed a lot of different algorithms in the past twenty years. One of the important applications of mesh parameterization is texturing which assigns a 2D image onto a 3D mesh surface, another one is remeshing.

Given a 3D mesh, the parameterization looks for a corresponding 2D flat mesh. The perfect mapping is an isometric one that can only exist on the developable surfaces. Therefore in practice, we try to preserve the area or angle. They are called authalic (area-preserving) mapping, conformal (angle-preserving) mapping, isometric (length-preserving) or some combination of them.

* Corresponding author at: DUT-RU International School of Information and Software Engineering, Dalian University of Technology, China.
E-mail address: nalei@outlook.com (N. Lei).

https://doi.org/10.1016/j.cagd.2018.03.001

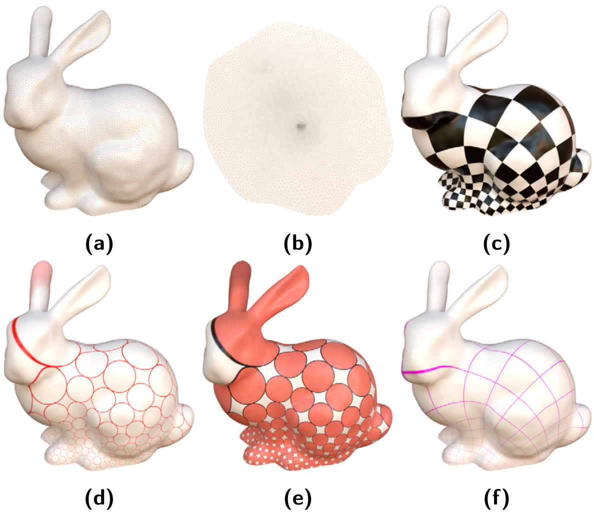

Fig. 1. (a) The bunny; (b) its parameterization with Calabi flow; (c), (d), (e) rendered with different textures.

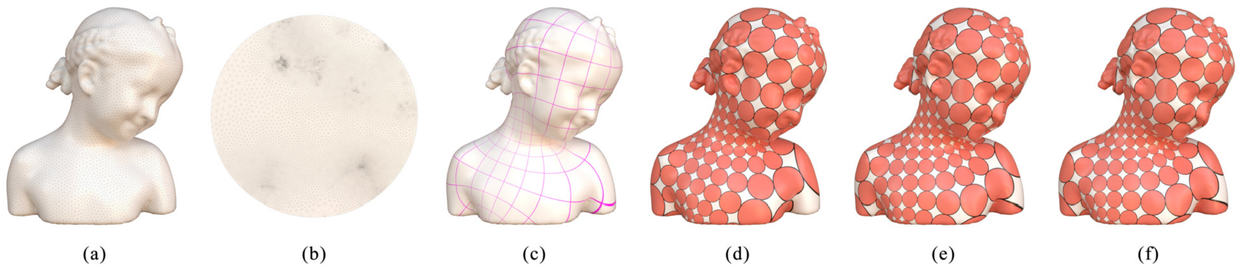

Fig. 2. (a) Original mesh; (b) the parameterization with Calabi flow; (c), (d) the texturing with Calabi flow; (e) the texturing with CETM; (f) the texturing with Ricci flow.

The algorithms proposed in Hormann and Greiner (2000), Fu et al. (2015) can be designed on the discrete triangle mesh directly. The optimal mapping (Desbrun et al., 2002a; Lévy et al., 2002; Liu et al., 2008) results from defining and minimizing an energy related to the mesh triangles. Other methodologies are based on the smooth surface mapping theories and then derive their corresponding discrete approximation (Jin et al., 2007a; Gu and Yau, 2003).

Flow-based algorithms do not work on the positions directly, instead they evolve the surface metric into a flat one. The final parametrization is obtained by embedding the surface of the flat metric to the 2D plane. In Fig. 1, we show our Calabi flow based conformal parameterization. The angles are preserved very well in several corresponding rendering results. In Fig. 2, we exhibit three mesh parameterizations with Calabi flow, Ricci flow and CETM.

Contribution. We design a different conformal parameterization algorithm based on discrete Calabi flow which is derived from the Calabi energy. In our algorithm, we use a new dual-Laplacian operator to obtain the optimal solution. To the best of our knowledge, this is the first time that the discrete Calabi flow and dual-Laplacian operator are introduced to the graphics literature. To summarize, our algorithm is a new perspective method to gain parameterization. The energy expression is simple and very easy to understand: it is squared difference between current curvature vector and target curvature. This is one of the main advantages towards Ricci flow and CETM.

2. Related works

Due to the abundance of literature on mesh parameterization. Here, we focus on approaches that are the most relevant to ours. We refer the reader to some excellent surveys (Sheffer et al., 2006, 2007) for more information.

Tutte's embedding. The algorithms (Floater, 2003; Desbrun et al., 2002b; Weber and Zorin, 2014; Tong et al., 2006) based on Tutte's embedding of planar graphs are fundamental ones. They map a 3D disk-topology meshes onto a Euclidean flat plane. The algorithm in Gortler et al. (2006) generalized them to handle genus-one meshes by integrating harmonic one-forms on the torus. Bright et al. (2017) generalized harmonic parameterization to arbitrary topology and Tong et al. (2006) use harmonic forms to generate quad mesh. Euclidean orbifolds are used to process sphere-topology meshes in Aigerman and Lipman (2015) which achieves flat surfaces with cone singularities. Recently orbifolds were also extended to hyperbolic space in Noam and Yaron (2016). Hyperbolic orbifolds are able to handle a wider variety of cone arrangements and topologies than Euclidean orbifolds. Aigerman et al. (2017) extended orbifolds again which can bijectively parameterize surfaces into spherical target domains called spherical orbifolds. The advantages of all of these methods are derived from that Tutte's theory and are guaranteed to be bijective.

Injective parameterization. Besides bijective, the injective and distortion bounded algorithms (Myles and Zorin, 2012, 2013; Myles et al., 2014) are also sought after. The algorithms presented in Hormann and Greiner (2000), Sheffer et al. (2005), Schüller et al. (2013), Aigerman et al. (2014), Weber and Zorin (2014), Fu et al. (2015), Diamanti et al. (2015), Chien et al. (2016) are locally injective, and the ones in Lipman (2012), Campen et al. (2015), Smith and Schaefer (2015) are globally injective.

Conformal parameterization. Based on conformal geometry theory, discrete conformal mapping definition is proposed. Ricci flow (Jin et al., 2007a, 2008a, 2008b), circle packing (Stephenson, 2005), circle patterns (Kharevych et al., 2006), conformal equivalence of triangle meshes (CETM) (Springborn et al., 2008) and conformal flattening (Ben-Chen et al., 2008) are presented. The relationships and comparisons among these methods are discussed in Zhang et al. (2014, 2015). All of these algorithms achieve a discrete flat metric under certain flows. They iteratively update the edge lengths which are conformal to the original mesh in each step. Ricci flow can also work under hyperbolic background geometry (Jin et al., 2006; Yang et al., 2009; Shi et al., 2013). However, these flow-based methods are not guaranteed to be injective.

Another approach for conformal parameterization is based on the conformal structure in Riemann surface theory. In the seminal paper (Gu and Yau, 2003), the discrete holomorphic differentials are defined. The conformal mapping is achieved by computing discrete conformal structures.

Area-preserving. Another kind of parameterizations is area-preserving. Recently discrete optimal mass transport theory is designed (Gu et al., 2013) and applied to obtain the map which can preserve the local triangle areas (Su et al., 2013, 2016; Zhao et al., 2013). The conformal and area-preserving approaches can also be mixed or interpolated by polar factorization method (Yu et al., 2018) to obtain the parameterization between them.

3. Calabi energy and Calabi flow

In the interest of being self-contained, we review some basic mathematical definitions and notations. For more detailed information on these topics, we refer to Lee (2003).

A smooth n-dimensional manifold M is a topological space which is locally Euclidean of dimension n and it can be covered by a series of coordinate charts $\{U_\alpha, \phi_\alpha\}$ which are C^∞ compatible (Lee, 2003). A Riemannian metric tensor $\mathbf{g}$ on the manifold is a Euclidean inner product defined on the tangent space $T_p(M^n)$ of each point p of M (Petersen, 2006).

A 2-dimensional surface is usually denoted as S, we usually embed it in $\mathbb{R}^3$, and equip each point with a local chart:

$$\mathbf{r}: U \to \mathbb{R}^3,$$

where $U \in \mathbb{R}^2$ and $\mathbf{r}$ is smooth. $\mathbf{r}$ is called a parameterization of S. At each point, let $\mathbf{r}_i = \partial \mathbf{r}/\partial u^i, i = 1, 2$ be the tangent vectors along the isoparametric curves. They are the basis of tangent space at that point. The length of a general tangent vector $d\mathbf{r} = \mathbf{r}_1 du_1 + \mathbf{r}_2 du_2$ can be computed by:

$$ds^2 = \langle d\mathbf{r}, d\mathbf{r}\rangle = \begin{pmatrix} du_1 & du_2 \end{pmatrix} \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} \begin{pmatrix} du_1 \\ du_2 \end{pmatrix}, \tag{1}$$

where $\langle -, - \rangle$ is the inner product in $\mathbb{R}^3$, and $g_{ij} = \langle \mathbf{r}_i, \mathbf{r}_j \rangle$. In this case, the matrix $\mathbf{g} = (g_{ij})$ is the Riemannian metric tensor on S. And we denote the inner product induced by $\mathbf{g}$ as $\langle -, - \rangle_{\mathbf{g}}$.

The angle between two tangent vectors can be measured by $\mathbf{g}$. Suppose $\delta\mathbf{r} = \mathbf{r}_1 \delta u_1 + \mathbf{r}_2 \delta u_2$ is another tangent vector, the angle between $d\mathbf{r}$ and $\delta\mathbf{r}$ measure by $\mathbf{g}$ is defined as:

$$\theta_{\mathbf{g}} = \cos^{-1} \frac{\langle d\mathbf{r}, \delta\mathbf{r}\rangle_{\mathbf{g}}}{\sqrt{\langle d\mathbf{r}, d\mathbf{r}\rangle_{\mathbf{g}}}\sqrt{\langle \delta\mathbf{r}, \delta\mathbf{r}\rangle_{\mathbf{g}}}}. \tag{2}$$

Suppose $\lambda : S \to \mathbb{R}$ is a real function defined on the surface. Define another Riemannian metric

$$\bar{\mathbf{g}} = e^{2\lambda}\mathbf{g},$$

then we have

$$\langle d\mathbf{r}, \delta\mathbf{r}\rangle_{\bar{\mathbf{g}}} = e^{2\lambda} \langle d\mathbf{r}, \delta\mathbf{r}\rangle_{\mathbf{g}}.$$

According to Eq. (2), we obtain $\theta_{\mathbf{g}} = \theta_{\bar{\mathbf{g}}}$. So we say $\mathbf{g}$ and $\bar{\mathbf{g}}$ is conformally equivalent and $e^{2\lambda}$ is conformal factor between $\bar{\mathbf{g}}$ and $\mathbf{g}$.

Every Riemannian metrics of 2-dimensional surface is **locally** conformally equivalent to the Euclidean flat metric (Chern, 1955). That is, we can always choose a set of special parameters, such that the metric is represented as:

$$ds^2 = e^{2\lambda}(du_1^2 + du_2^2).$$

Such kinds of parameterizations are called the isothermal coordinates of the surface.

Under isothermal coordinates, the Gaussian curvature is represented as:

$$K = -e^{-2\lambda}\nabla^2\lambda,$$

where ∇^2 is the normal Laplace operator:

$$\nabla^2 = \frac{\partial^2}{\partial u_1^2} + \frac{\partial^2}{\partial u_2^2},$$

and $e^{-2\lambda}\nabla^2$ is called the Laplace–Beltrami operator. Here we denote it as Δ.

2-dimensional Calabi flow was studied in Chen and He (2008). Suppose S is a smooth surface with a Riemann metric $\mathbf{g}$, Calabi introduced the so-called Calabi energy, which is defined as:

$$\Phi(\mathbf{g}) = \int_S K^2 dA, \tag{3}$$

where dA is the area element of S.

The Calabi flow on S is defined as:

$$\frac{dg_{ij}}{dt} = 2\Delta K g_{ij} \tag{4}$$

where K is the Guassian curvature induced by metric $\mathbf{g}$ which is equal to $2K$ and Δ is the Laplace–Beltrami operator.

With isothermal coordinates, we have $\mathbf{g} = e^{2\lambda}\mathbf{g}_0$, then the Calabi flow becomes:

$$\frac{d\lambda}{dt} = \Delta K. \tag{5}$$

It is proved that the above Calabi flow is convergent under certain conditions (Calabi, 1982).

4. Discrete metric and conformal class

In practice, smooth surfaces are often approximated by simplicial complexes, that is, piecewise linear triangle meshes. Concepts in the continuous setting can be generalized to the discrete setting. In this paper, a triangle mesh is denoted as M, which is associated with a vertex set V, an edge set E and a face set F. Here v_i represents a certain vertex, e_{ij} represents the edge between vertices v_i and v_j, and f_{ijk} represents the face formed by v_i, v_j and v_k.

A Riemannian metric on a piecewise linear discrete mesh $M = (V, E, F)$ is defined as a positive scalar function on edges:

$$l : E \to \mathbb{R}^+,$$

such that for each triangle face f_{ijk}, edge lengths $\{l_{ij}, l_{jk}, l_{ki}\}$ satisfy the triangle inequality:

$$l_{ij} + l_{jk} > l_{ki}; \quad l_{ij} + l_{ki} > l_{jk}; \quad l_{jk} + l_{ki} > l_{ij}.$$

The discrete metric determines corner angles $\{\theta_i^{jk}, \theta_j^{ki}, \theta_k^{ij}\}$ of triangles by cosine laws in Euclidean background geometries.

In discrete setting, we have infinite ways to assign edge lengths to make triangle inequalities be satisfied. However, it is difficult to settle down what the conformal deformation means. Inspired by the property that conformal map sends infinitesimal circles to infinitesimal circles, Thurston introduced circle packing on weighted meshes in Thurston (1976). Inspired by conformal factors, Luo introduced another metric in Luo (2004) called Yamabe flow metric. And as a generalization of Thurston's circle packing metric, inverse distance metric was introduced in Yang et al. (2009). These two metrics induced different definitions of conformal class. Both are approximations of the definition of conformal deformation in smooth settings. In Zhang et al. (2014), a unified framework for circle packing was introduced and the metrics above were all included.

Thurston's circle packing metric. The circle packing metric was firstly defined on weighted meshes in Thurston (1976). A weighted mesh (M, Γ, Φ) with a circle packing metric is a mesh with a function Γ assigning a radius r_i to each vertex v_i:

$$\Gamma : V \to \mathbb{R}^+,$$

and a function assigning a weight Φ_{ij} to each edge:

$$\Phi : E \to [0, \frac{\pi}{2}].$$

Under this setting, we can determine edge lengths using different cosine laws in different background geometry:

$$l_{ij}^2 = r_i^2 + r_j^2 + 2r_i r_j \cos\Phi_{ij}.$$

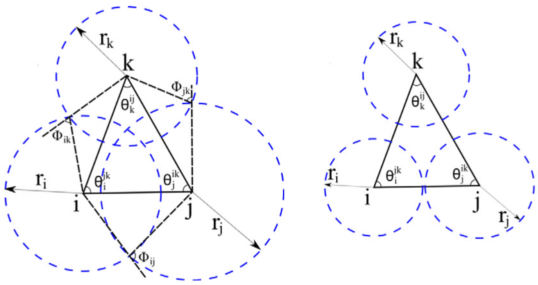

Fig. 3. The two kinds of metrics. (a) Thurston's circle packing, (b) inversive distance circle packing.

Given a mesh M, we say two circle metrics (Γ_1, Φ_1) and (Γ_2, Φ_2) are conformally equivalent if $\Phi_1 = \Phi_2$, they are in the same conformal class.

Inversive distance circle packing metric. This kind of metric is first introduced in Bowers and Stephenson (2004). It is a generalization of Thurston's circle packing. We define a function on edges $I : E \to \mathbb{R}$, which is called inversive distance function. Edge lengths is determined as follows:

$$l_{ij}^2 = r_i^2 + r_j^2 + 2 r_i r_j I_{ij} .$$

Given a mesh M, we say two circle metrics (Γ_1, I_1) and (Γ_2, I_2) are conformally equivalent if $I_1 = I_2$, they are in the same conformal class. This kind of metric can approximate initial edge lengths very well, so it is more practical than Thurston's circle packing.

Geometric interpretation. Two kinds of circle packing metrics can be illustrated as Fig. 3. Each vertex v_i has a circle with radius r_i centering at it. For Thurston's circle packing metric, two adjacent circles intersect with angle Φ_{ij}. The edge length is the distance between two circle centers. For inverse distance circle metric, the circles need not intersect with each other, and $\cos \Phi_{ij}$ is replaced by the inverse distance I_{ij}.

5. Our algorithm

Discrete Calabi flow (Ge, 2018) defined on triangular meshes is an approximation of smooth Calabi flow on smooth surfaces. Given weighted mesh (M, Γ, Φ) with circle packing metric, we set:

$$u_i = \log r_i .$$

We define discrete Calabi flow as:

$$\frac{d\mathbf{u}}{dt} = \Delta_{dual} \mathbf{K}, \tag{6}$$

where Δ_{dual} is a new kind of Laplacian operator, we call it discrete **dual-Laplacian operator** (Ge, 2018), which is defined as:

$$\Delta_{dual} = -L_{dual} = -\nabla_u K = (L_{ij})_{N\times N} = -\frac{\partial(K_1, \ldots, K_N)}{\partial(u_1, \ldots, u_N)} \tag{7}$$

$$= -\begin{pmatrix} \frac{\partial K_1}{\partial u_1} & \cdot & \cdot & \cdot & \frac{\partial K_1}{\partial u_N} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \frac{\partial K_N}{\partial u_1} & \cdot & \cdot & \cdot & \frac{\partial K_N}{\partial u_N} \end{pmatrix},$$

and $\mathbf{K}$ is the well-known traditional Gaussian curvature which is computed as the angle deficit, i.e. 2π minus angle sum for an inner vertex and π minus angel sum for a boundary vertex.

We can also modify discrete Calabi flow into a form with prescribed curvature:

$$\frac{d\mathbf{u}}{dt} = \Delta_{dual} (\mathbf{K} - \bar{\mathbf{K}}) \tag{8}$$

and $\bar{\mathbf{K}} = (\bar{K_1}, \bar{K_2}, \ldots, \bar{K_N})$ is the prescribed curvature vector.

Calabi flow is similar to Ricci flow. To compare, we notice that the Ricci flow is expressed by the following format:

$$\frac{d\mathbf{u}}{dt} = \mathbf{K} - \bar{\mathbf{K}}.$$

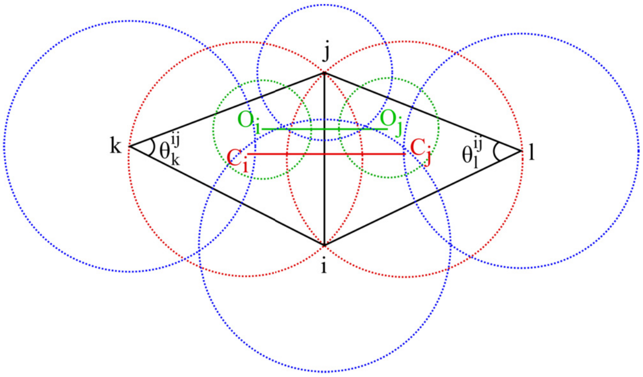

Fig. 4. Geometric interpretation of dual Laplacian and cotangent Laplacian.

The discrete Calabi energy is defined as the following.

$$\mathbf{C}(u) = \sum_{v_i \in V} (\bar{K}_i - K_i)^2.$$

The discrete Calabi flow is the negative gradient flow of discrete Calabi energy, and the discrete Calabi energy is descending along this flow (Ge, 2018). With a little calculation, we can find that

$$\nabla_u \mathbf{C} = \Delta_{dual}(\bar{\mathbf{K}} - \mathbf{K})$$

If Calabi flow converges, Calabi energy will arrive at its critical point, then:

$$\Delta_{dual}(\bar{\mathbf{K}} - \mathbf{K}) = 0.$$

Due to the results of Guo (2011), L_{dual} is positive defined, so after the flow converges, we obtain a metric with prescribed curvatures.

Dual-Laplacian operator. The dual-Laplacian operator Δ_{dual} is different from the well-known cotangent Laplacian operator Δ_{cot}. It is a special type of the discrete Laplacian and comes from the dual structure of the circle packings (Goes et al., 2014). It has been discussed and studied by Glickenstein (2005a, 2005b) systematically.

Both Δ_{dual} and Δ_{cot} operate on the column vector functions which are defined on mesh vertices with a matrix multiplication.

We need to derive the explicit form of $L_{dual} = \nabla_u \mathbf{K}$. The calculation is direct. We write $i \sim j$ in the following if v_i and v_j are adjacent.

If $i \sim j$,

$$(L_{dual})_{ij} = (L_{dual})_{ji} = \frac{\partial K_i}{\partial u_j} = -\sum_{f_{ijk}} \frac{\partial \theta_i^{jk}}{\partial u_j}. \tag{9}$$

If $i = j$, according to Gauss–Bonnet Theorem, we have

$$(L_{dual})_{ii} = \frac{\partial K_i}{\partial u_i} = -\sum_{j \sim i} \frac{\partial K_j}{\partial u_i}, \tag{10}$$

otherwise,

$$(L_{dual})_{ij} = 0. \tag{11}$$

So the calculation of dual Laplacian boils down to the calculation of $(L_{dual})_{ij}$. This quantity can be associated to edge e_{ij}. We call it edge weight, and denote it as $(w_d)_{ij}$, which is only determined by its two adjacent faces.

For the equation (9), there is a formula with nice geometric interpretation. As shown in Fig. 4, the face f_{ijk} and face f_{ijl} are two adjacent faces at edge e_{ij}. On each vertex of the face, there is a circle with the radius value of its corresponding metric. For each face of the triangle, we can assign a circle orthogonal to three vertex circles simultaneously. This circle is called the power circle of the face and its center is called power center. It can be shown that the line connecting two power centers O_i and O_j in this figure is orthogonal to edge e_{ij}, and we call this segment as dual edge of e_{ij}. Denote the length of e_{ij} as l_{ij} and the length of its dual edge as $l_{ij}^o = |O_i O_j|$.

Finally we have a very nice formula for edge weight $(w_d)_{ij}$:

$$(L_{dual})_{ij} = (w_d)_{ij} = -\frac{\partial(\theta_i^{jk} + \theta_i^{jl})}{\partial u_j} = \frac{l_{ij}^o}{l_{ij}}. \tag{12}$$

On the other hand, the formula of the well-known cotangent-Laplacian operator in computer graphics community is expressed as:

$$\Delta_{cot} = -L_{cot}^T.$$

The operator L_{cot} is computed as follows:
If $i \sim j$,

$$(L_{cot})_{(ij)} = (L_{cot})_{ji} = \frac{1}{2}\sum_{f_{ijk}} \cot\theta_k^{ij}.$$

If $i = j$,

$$(L_{cot})_{ii} = -\sum_{j\sim i}(L_{cot})_{ij},$$

otherwise,

$$(L_{cot})_{ij} = 0.$$

The computation of the cotangent Laplacian is related to the corresponding edge weights. And it has an geometric interpretation too. As shown in Fig. 4, there is a circumcircle for every triangle, the line segment between two circumcenter C_i and C_j is also called dual edge of e_{ij}. Denote the length of e_{ij} as l_{ij} and length of its dual edge as l_{ij}^c, then we have

$$(L_{cot})_{ij} = (w_c)_{ij} = \frac{\cot\theta_k^{ij} + \cot\theta_l^{ij}}{2} = \frac{l_{ij}^c}{l_{ij}}.$$

Actually, the circumcenter can be also treated as some kind of power center. In this case, three vertex circles shrink to a point and the orthogonal circle of them is exactly the same as the circumcircle. Therefore these two kinds of Laplacian operators have unified forms.

Algorithm 1 Compute initial inversive distance circle packing metric.

```
for e_ij ∈ E do
    d_ij ← the original edge length of e_ij in ℝ^3
end for
for f_ijk ∈ F do
    r_i^jk ← (d_ki + d_ij − d_jk) / 2
end for
for v_i ∈ V do
    Determine vertex radius by: r_i = min_{f_ijk} r_i^jk
end for
for e_ij ∈ E do
    compute edge metric weight by I_ij ← (d_ij^2 − r_i^2 − r_j^2) / (2 r_i r_j)
end for
```

As the same with discrete Ricci flow, the solution exists and converges if and only if it satisfies Thurston's circle packing condition which is explained in detail in Ge (2018). The convergent analysis of discrete Calabi flow for circle packing metric and inverse distance metric are discussed in Ge (2018) and Ge and Jiang, (2017a, 2017b) respectively.

Initial metric. The Calabi flow starts with an initial metric, there are some choices, such as Thurston's circle packing, inverse distance circle packing, and so on Springborn et al. (2008), Zhang et al. (2014). The desired metric should approximate original edge lengths in $\mathbb{R}^3$ as much as possible. Inverse distance circle packing metric is equal to the original edge length, and Thurston's circle packing can only approximate them. In our experiments, inversive distance circle packing metric does well in terms of this aspect. Therefore we use the inversive distance circle packing metric in our experiments. We follow the method in Yang et al. (2009) to compute the initial inverse distance circle packing metric. The detail of the algorithms is shown in Algorithm 1.

In Fig. 5, we show the comparison of our Calabi flow based conformal parameterization by Thurston's circle packing metric and inverse distance circle packing metric, we observe that inverse distance metric has better conformal results when meshes are coarse.

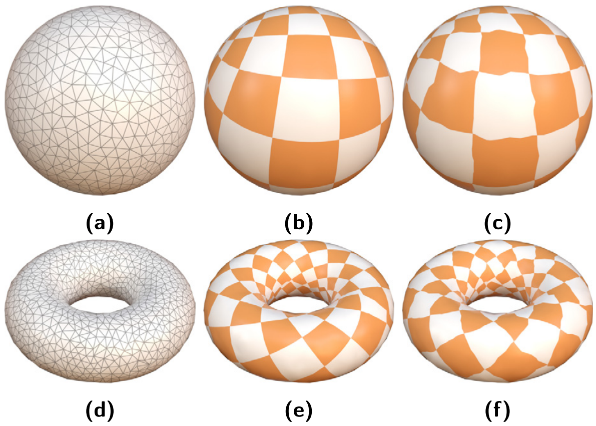

Fig. 5. (a), (d) The original meshes; (b), (e) conformal parameterization with inverse distance circle packing metric; (c), (f) conformal parameterization with Thurston's circle packing metric.

Gradient descending. Calabi energy is minimized to obtain the optimal solution. We use the method of gradient descending to solve the optimization problem of Calabi energy. The scheme of conjugate gradient descending and accelerated gradient descending can also be applied. Intuitively, we can treat $\frac{\partial K_j}{\partial u_i}(\bar{K}_j - K_j)$ as the descending direction of K_j along axis u_i. So the gradient of Calabi energy can be seen as the average of adjacent descending directions. The whole procedure of the gradient descending algorithm is shown in Algorithm 2.

Algorithm 2 Calabi flow.

Compute an initial circle packing metric.
Set target curvatures of each vertex.
while $\max_i |K_i - \bar{K}_i| < \epsilon$ **do**
    Calculate curvatures according current metric.
    Calculate dual Laplacian L.
    Compute the updating direction $d\mathbf{u} \leftarrow L^T(\bar{\mathbf{K}} - \mathbf{K})$.
    Update conformal factors of each vertex by $\mathbf{u} \leftarrow \mathbf{u} + \delta d\mathbf{u}$.
end while ▷ This procedure is the optimization of Calabi energy.
Embed the mesh to Euclidean plane.

Embedding. After the Calabi flow converges, we obtain a flat metric. The 2D vertex positions which are compatible with the metric need to be calculated. We choose a triangle face as root and embed it onto Euclidean plane, then we use the breadth-first method to embed other triangle faces. The details of our algorithm are shown in Algorithm 3.

Algorithm 3 Embed the mesh with flat metric to Euclidean plane.

Choose a root face and then embed it.
for f_{ijk} in the sequence of breadth-first search of all faces **do**
    if all vertices of f_{ijk} have been embedded **then**
        return
    else
        Compute intersections of circles $C(p_i, l_{ik})$ and $C(p_j, l_{jk})$.
        Choose the intersection which preserve the orientation of f_{ijk} as the embedding of v_k. ▷ We assume v_i and v_j have been embedded.
    end if
end for

6. Experiments

According to Gauss–Bonnet theorem, only genus one closed surfaces admit Euclidean conformal structure, which means a flat metric without any singularity. In this setting, we set all target curvatures as zero, and run flow on it directly. When the flow converges, we slice the mesh into a disk and then the mesh can be embedded into Euclidean plane. In Fig. 6, we show the parameterizations of the genus one meshes.

For the closed meshes with genus but one, we need cut them into the disk-type topology. We design experiments with three different kinds of boundary settings: the boundary with fixed curvatures, circle boundary, and free boundary.

Fixed boundary. The result of this kind of configuration is a polygon with corner angles being the π minus target curvature $\bar{K}_i$. In the step of setting target curvatures, we set all target curvatures of each interior vertices zero, and all target

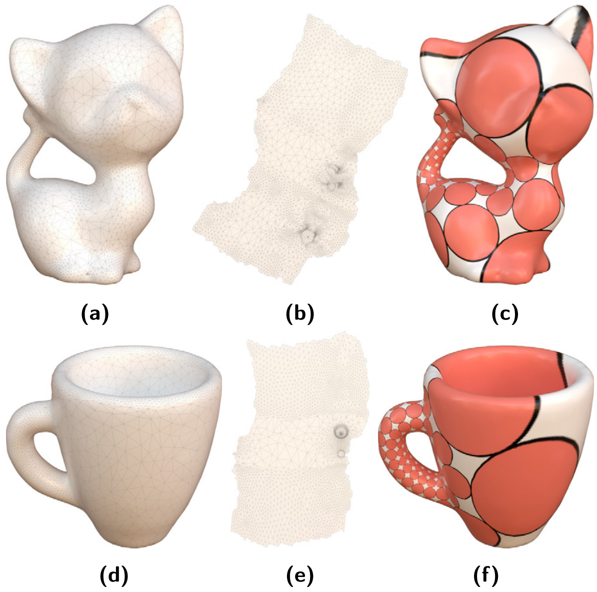

Fig. 6. The meshes of genus one and their Calabi flow based parameterizations.

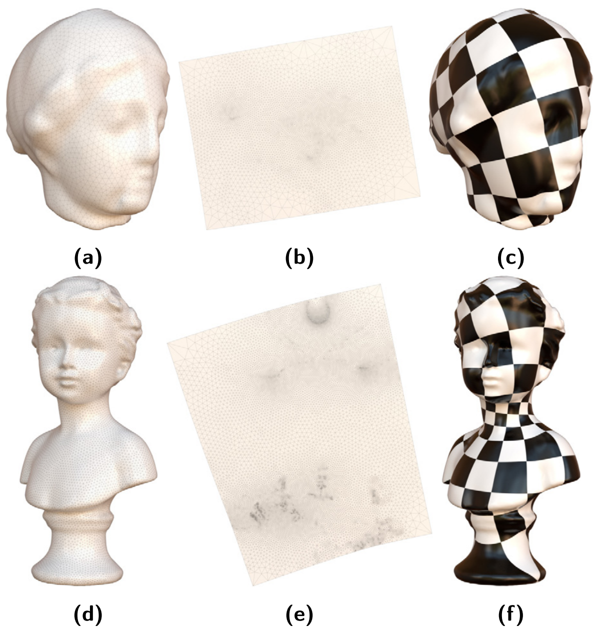

Fig. 7. The Calabi flow based parametrizations with fixed rectangle boundary.

curvatures of boundary vertices zero except corner vertices. We should guarantee that the sum of target curvatures of these corner vertices equals to 2π, make it admissible with Gauss–Bonnet Theorem.

In Fig. 7, we show two parametrization results with fixed rectangle boundaries.

Circular boundary. If the target boundary is a circle, we cannot simply set all boundary target curvatures to be $\frac{2\pi}{m}$, where m is the number of vertices of the boundary. In fact, if the boundary is a circle, the curvature of each boundary vertex should satisfy the following conditions. And we must update these target curvatures in each iteration.

$$\frac{K_i}{l_{i-1,i}+l_{i,i+1}} \equiv c, \forall v_i \in \partial M,$$

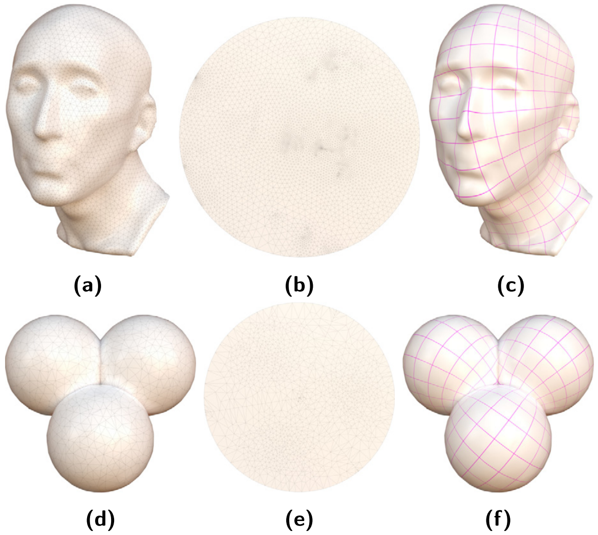

Fig. 8. The Calabi flow based parametrization results with the circle boundaries.

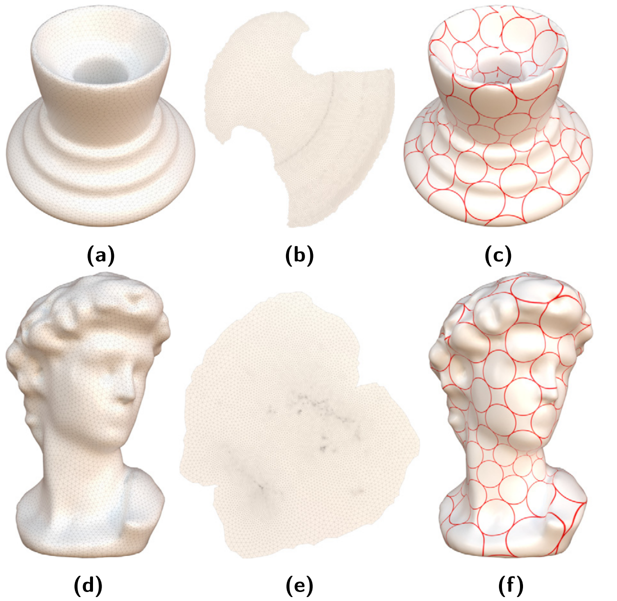

Fig. 9. Calabi flow based parameterizations with free boundary.

where l_{ij} is edge length under the target metric, and

$$\sum_{v_i \in \partial M} K_i = 2\pi \ .$$

We demonstrate two Calabi flow conformal parameterization results with circle boundaries in Fig. 8.

Free boundary. In this setting, we do not set boundary target curvatures directly, instead, we set $d\mathbf{u}$ of boundary vertices in Algorithm 2 to be zero. In this way, the Calabi energy will continue to decrease, and this will lead to the smaller area distortion. To be specific, we set the value of $d\mathbf{u}$ on the boundary be zero, and only update the interior vertices. Some results are shown in Fig. 9.

We run experiments on dozens of models and compare the conformities of our Calabi flow, Ricci flow, and CETM algorithms. In Fig. 13, we show several meshes parametrized with Calabi flow, Ricci flow, and CETM algorithms respectively. We observed that three kinds of algorithms have almost the same conformities and all of them can preserve the angles nicely.

We calculate the statistic of angle error ratios of the original kitten model and its parameterizations by three kinds of algorithms. In Fig. 10, we show that three methods exhibit almost the same distribution of angle ratios. We choose 300

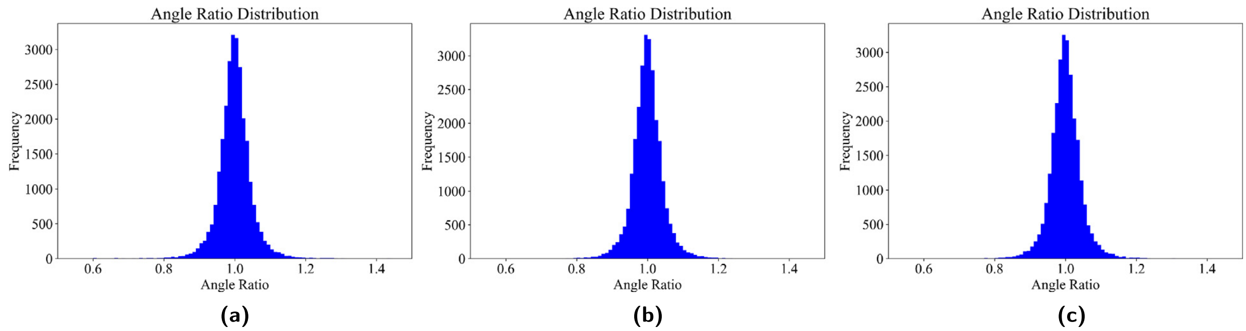

Fig. 10. Angle error ratio distributions of the kitten model by Calabi flow (a), Ricci flow (b) and CETM (c) respectively.

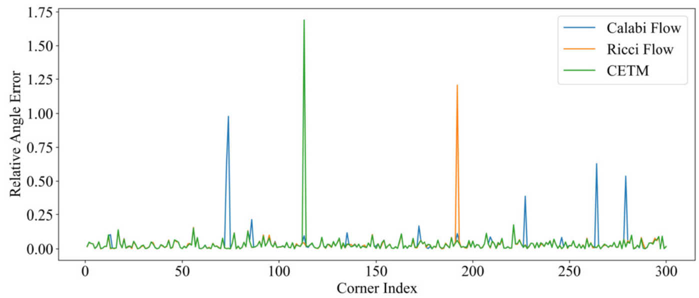

Fig. 11. Angle relative errors on randomly selected 300 faces of kitten model.

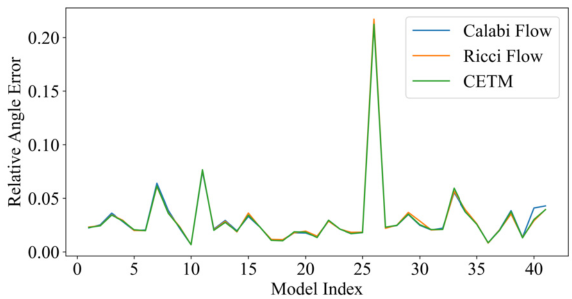

Fig. 12. Mean angle relative errors on all models.

corners of the kitten model randomly. The relative angle errors are shown in Fig. 11. It is observed that there is some local conformal difference of three algorithms. Finally, we exhibit the mean relative angle errors of all test models in Fig. 12. Based on these experiments, we conclude that our Calabi flow method has the same conformity with Ricci flow and CETM.

In our experiments, as the same as Ricci flow and CETM, Calabi flow may fail on the meshes with very narrow triangles or with large curvatures. It means that the convergence of these discrete flows depend on the mesh quality.

7. Conclusions

In this paper, we present a conformal mesh parameterization based on discrete Calabi flow. This flow is different from previous well-known Ricci flow and CETM, which can produce comparable results to the other two flows. The strongest aspect of Calabi flow is the simplicity of the optimization energy, while Ricci flow and CETM's energies cannot be easily recognized.

Currently, we only use Calabi flow on the Euclidean background. It is also possible to explore Calabi flow in the hyperbolic background (Ge and Xu, 2016).

In Ricci flow, dynamic edge flips (Luo, 2004) can theoretically be employed for meshes of low quality to enable the convergence of the flow. It is also possible to exploit this technique to Calabi flow in future.

Calabi energy has a clear geometric meaning. It expressed the difference between current Gaussian curvature and target Gaussian curvature. It is potential to exploit this feature in mesh smoothing and other related operations.

Calabi flow has higher order than Ricci flow. The convergence of the flow is proved under Thurston's circle packing metric in Ge (2018). In our experiments, we use the inversive distance circle packing metric, which improves the quality

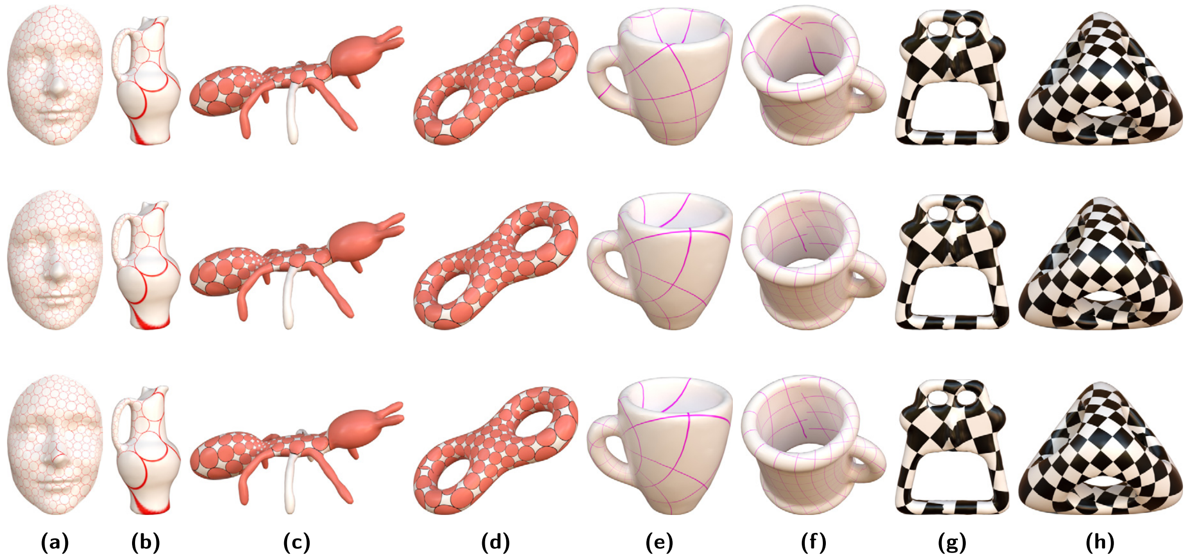

Fig. 13. The first row is with Calabi flow; the second is with Ricci flow; The third is with CETM.

a lot. However, the rigorous proof of the convergence under this kind of metric has not been settled yet. Since the main purpose of this paper is to introduce Calabi flow to the graphics community, we have only tried the method and simply used the gradient descent. It is potential to exploit faster optimization method and prove the convergence under inversive distance circle packing metric to get insight in the robustness of Calabi flow in the future.

The conformal flow tool, such as Ricci flow, has already been used in the applications of brain analysis (Wang et al., 2007), shape analysis (Gu et al., 2007), network routing (Sarkar et al., 2009; Gao et al., 2016), shape modeling (Patanè et al., 2014), shape registration (Gao et al., 2013), face matching (Zeng et al., 2008), geometric structures (Jin et al., 2007b), and so on. In the future, we will explore the applicability of Calabi flow in these kinds of applications.

Acknowledgements

We would like to thank anonymous reviewers for their insightful feedbacks, valuable comments, and suggestions. Some pictures are rendered by Mitsuba (Jakob, 2010). All 3D models are from the AIM@SHAPE shape repository and Thingi10K repository. Thanks MeshDGP (Zhao, 2016) framework for the implementation reference. The project is partially supported by NSFC No. 61772105, 61720106005, NSF DMS-1418255, AFOSR FA9550-14-1-0193, NIH 1R01LM012434. Ge is supported by NSFC under grant No. 11501027. Wang is supported by NSFC No. 11772047 and Key International Collaborating Project from NSFC No. 11620101001.

References

Aigerman, N., Kovalsky, S.Z., Lipman, Y., 2017. Spherical orbifold Tutte embeddings. ACM Trans. Graph. 36 (4), 90.
Aigerman, N., Lipman, Y., 2015. Orbifold Tutte embeddings. ACM Trans. Graph. 34, 190.
Aigerman, N., Poranne, R., Lipman, Y., 2014. Lifted bijections for low distortion surface mappings. ACM Trans. Graph. 33, 69.
Ben-Chen, M., Gotsman, C., Bunin, G., 2008. Conformal flattening by curvature prescription and metric scaling. In: Computer Graphics Forum, vol. 27. Wiley Online Library, pp. 449–458.
Bowers, P.L., Stephenson, K., 2004. Uniformizing Dessins and Belyi Maps Via Circle Packing, vol. 170. American Mathematical Soc.
Bright, A., Chien, E., Weber, O., 2017. Harmonic global parametrization with rational holonomy. ACM Trans. Graph. 36 (4), 89.
Calabi, E., 1982. Extremal Kähler Metrics. Annals of Mathematics Studies, pp. 259–290.
Campen, M., Bommes, D., Kobbelt, L., 2015. Quantized global parametrization. ACM Trans. Graph. 34, 192.
Chen, X., He, W., 2008. On the Calabi flow. Am. J. Math. 130 (2), 539–570.
Chern, S.-S., 1955. An elementary proof of the existence of isothermal parameters on a surface. Proc. Am. Math. Soc. 6 (5), 771–782.
Chien, E., Levi, Z., Weber, O., 2016. Bounded distortion parametrization in the space of metrics. ACM Trans. Graph. (TOG) 35 (6), 215.
Chow, B., Luo, F., et al., 2003. Combinatorial Ricci flows on surfaces. J. Differ. Geom. 63 (1), 97–129.
Desbrun, M., Meyer, M., Alliez, P., 2002a. Intrinsic parameterizations of surface meshes. In: Computer Graphics Forum, vol. 21. Wiley Online Library, pp. 209–218.
Desbrun, M., Meyer, M., Alliez, P., 2002b. Intrinsic parameterizations of surface meshes. Comput. Graph. Forum 21, 209–218.
Diamanti, O., Vaxman, A., Panozzo, D., Sorkine-Hornung, O., 2015. Integrable polyvector fields. ACM Trans. Graph. (TOG) 34 (4), 38.
Floater, M.S., 2003. One-to-one piecewise linear mappings over triangulations. Math. Comput. 72, 685–696.
Fu, X.-M., Liu, Y., Guo, B., 2015. Computing locally injective mappings by advanced MIPS. ACM Trans. Graph. (TOG) 34 (4), 71.
Gao, J., Gu, X., Luo, F., 2016. Discrete Ricci flow for geometric routing. In: Encyclopedia of Algorithms.
Gao, M., Shi, R., Zhang, S., Zeng, W., Qian, Z., Gu, X., Metaxas, D.N., Axel, L., 2013. High resolution cardiac shape registration using Ricci flow. In: ISBI.
Ge, H., 2012. Combinatorial Methods and Geometric Equations. Thesis (Ph.D.). Peking University, Beijing.

Ge, H., 2018. Combinatorial Calabi flows on surfaces. Trans. Am. Math. Soc. 370 (2), 1377–1391.
Ge, H., Jiang, W., 2017a. On the deformation of inversive distance circle packings, II. J. Funct. Anal. 272 (9), 3573–3595.
Ge, H., Jiang, W., 2017b. On the deformation of inversive distance circle packings, III. J. Funct. Anal. 272 (9), 3596–3609.
Ge, H., Xu, X., 2016. 2-dimensional combinatorial Calabi flow in hyperbolic background geometry. Differ. Geom. Appl. 47, 86–98.
Glickenstein, D., 2005a. A combinatorial Yamabe flow in three dimensions. Topology 44 (4), 791–808.
Glickenstein, D., 2005b. Geometric triangulations and discrete laplacians on manifolds. Preprint. arXiv:math/0508188.
Goes, F.d., Memari, P., Mullen, P., Desbrun, M., 2014. Weighted triangulations for geometry processing. ACM Trans. Graph. 33 (3), 28.
Gortler, S.J., Gotsman, C., Thurston, D., 2006. Discrete one-forms on meshes and applications to 3D mesh parameterization. Comput. Aided Geom. Des. 23, 83–112.
Gu, X., Luo, F., Sun, J., Yau, S.-T., 2013. Variational principles for Minkowski type problems, discrete optimal transport, and discrete Monge–Ampere equations. Preprint. arXiv:1302.5472.
Gu, X., Wang, S., Kim, J., Zeng, Y., Wang, Y., Qin, H., Samaras, D., 2007. Ricci flow for 3D shape analysis. In: ICCV.
Gu, X., Yau, S.-T., 2003. Global conformal surface parameterization. In: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing. Eurographics Association, pp. 127–137.
Guo, R., 2011. Local rigidity of inversive distance circle packing. Trans. Am. Math. Soc. 363 (9), 4757–4776.
Hormann, K., Greiner, G., 2000. Mips: An Efficient Global Parametrization Method. Technical report, DTIC Document.
Jakob, W., 2010. Mitsuba renderer. http://www.mitsuba-renderer.org.
Jin, M., Kim, J., Gu, X., 2007a. Discrete surface Ricci flow: theory and applications. In: IMA Conference on the Mathematics of Surfaces.
Jin, M., Kim, J., Luo, F., Gu, X., 2008a. Discrete surface Ricci flow. IEEE Trans. Vis. Comput. Graph. 14, 1030–1043.
Jin, M., Kim, J., Luo, F., Gu, X., 2008b. Variational method on discrete Ricci flow. In: IWCIA Special Track on Applications.
Jin, M., Luo, F., Gu, X., 2006. Computing surface hyperbolic structure and real projective structure. In: Proceedings of the 2006 ACM Symposium on Solid and Physical Modeling. ACM, pp. 105–116.
Jin, M., Luo, F., Gu, X., 2007b. Computing general geometric structures on surfaces using Ricci flow. Comput. Aided Des. 39, 663–675.
Kharevych, L., Springborn, B., Schröder, P., 2006. Discrete conformal mappings via circle patterns. ACM Trans. Graph. (TOG) 25 (2), 412–438.
Lee, J.M., 2003. Introduction to Smooth Manifolds. Springer.
Lévy, B., Petitjean, S., Ray, N., Maillot, J., 2002. Least squares conformal maps for automatic texture atlas generation. In: ACM Transactions on Graphics (TOG), vol. 21. ACM, pp. 362–371.
Lipman, Y., 2012. Bounded distortion mapping spaces for triangular meshes. ACM Trans. Graph. 31, 108.
Liu, L., Zhang, L., Xu, Y., Gotsman, C., Gortler, S.J., 2008. A local/global approach to mesh parameterization. In: Computer Graphics Forum, vol. 27. Wiley Online Library, pp. 1495–1504.
Luo, F., 2004. Combinatorial Yamabe flow on surfaces. Commun. Contemp. Math. 6 (05), 765–780.
Myles, A., Pietroni, N., Zorin, D., 2014. Robust field-aligned global parametrization. ACM Trans. Graph. (TOG) 33 (4), 135.
Myles, A., Zorin, D., 2012. Global parametrization by incremental flattening. ACM Trans. Graph. (TOG) 31 (4), 109.
Myles, A., Zorin, D., 2013. Controlled-distortion constrained global parametrization. ACM Trans. Graph. (TOG) 32 (4), 105.
Noam, A., Yaron, L., 2016. Hyperbolic orbifold Tutte embeddings. ACM Trans. Graph. 35, 217.
Patanè, G., Li, X., Gu, X., 2014. An introduction to Ricci flow and volumetric approximation with applications to shape modeling. In: SIGGRAPH ASIA Courses.
Petersen, P., 2006. Riemannian Geometry. Springer-Verlag, New York.
Sarkar, R., Yin, X., Gao, J., Luo, F., Gu, X., 2009. Greedy routing with guaranteed delivery using Ricci flows. In: IPSN.
Schüller, C., Kavan, L., Panozzo, D., Sorkine-Hornung, O., 2013. Locally injective mappings. In: Computer Graphics Forum, vol. 32. Wiley Online Library, pp. 125–135.
Sheffer, A., Hormann, K., Levy, B., Desbrun, M., Zhou, K., Praun, E., Hoppe, H., 2007. Mesh parameterization: theory and practice.
Sheffer, A., Lévy, B., Mogilnitsky, M., Bogomyakov, A., 2005. Abf++: fast and robust angle based flattening. ACM Trans. Graph. 24, 311–330.
Sheffer, A., Praun, E., Rose, K., 2006. Mesh parameterization methods and their applications. Found. Trends Comput. Graph. Vis. 2.
Shi, R., Zeng, W., Su, Z., Damasio, H., Lu, Z., Wang, Y., Yau, S.-T., Gu, X., 2013. Hyperbolic harmonic mapping for constrained brain surface registration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2531–2538.
Smith, J., Schaefer, S., 2015. Bijective parameterization with free boundaries. ACM Trans. Graph. 34, 70.
Springborn, B., Schröder, P., Pinkall, U., 2008. Conformal equivalence of triangle meshes. In: ACM Transactions on Graphics (TOG), vol. 27. ACM, p. 77.
Stephenson, K., 2005. Introduction to Circle Packing: The Theory of Discrete Analytic Functions. Cambridge University Press.
Su, K., Cui, L., Qian, K., Lei, N., Zhang, J., Zhang, M., Gu, X.D., 2016. Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation. Comput. Aided Geom. Des. 46, 76–91.
Su, Z., Zeng, W., Shi, R., Wang, Y., Sun, J., Gu, X., 2013. Area preserving brain mapping. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2235–2242.
Thurston, W., 1976. Geometry and Topology of Three-Manifolds. Princeton Lecture Notes.
Tong, Y., Alliez, P., Cohen-Steiner, D., Desbrun, M., 2006. Designing quadrangulations with discrete harmonic forms. In: Proceedings of the Fourth Eurographics Symposium on Geometry Processing, pp. 201–210.
Wang, Y., Gu, X., Chan, T.F., Thompson, P.M., Yau, S.-T., 2007. Brain surface conformal parameterization with the Ricci flow. In: ISBI.
Weber, O., Zorin, D., 2014. Locally injective parametrization with arbitrary fixed boundaries. ACM Trans. Graph. 33, 75.
Yang, Y.-L., Guo, R., Luo, F., Hu, S.-M., Gu, X., 2009. Generalized discrete Ricci flow. Comput. Graph. Forum 28, 2005–2014.
Yu, X., Lei, N., Zheng, X., Gu, X., 2018. Surface parameterization based on polar factorization. J. Comput. Appl. Math. 329 (C), 24–36.
Zeng, W., Yin, X., Zeng, Y., Lai, Y.-K., Gu, X., Samaras, D., 2008. 3D face matching and registration based on hyperbolic Ricci flow. In: CVPR Workshops.
Zhang, M., Guo, R., Zeng, W., Luo, F., Yau, S.-T., Gu, X., 2014. The unified discrete surface Ricci flow. Graph. Models 76, 321–339.
Zhang, M., Zeng, W., Guo, R., Luo, F., Gu, X.D., 2015. Survey on discrete surface Ricci flow. J. Comput. Sci. Technol. 30, 598–613.
Zhao, H., 2016. MeshDGP: A C Sharp mesh processing framework. http://meshdgp.github.io/.
Zhao, X., Su, Z., Gu, X.D., Kaufman, A., Sun, J., Gao, J., Luo, F., 2013. Area-preservation mapping using optimal mass transport. IEEE Trans. Vis. Comput. Graph. 19 (12), 2838–2847.