J Sci Comput (2014) 58:705–725
DOI 10.1007/s10915-013-9752-6

Folding-Free Global Conformal Mapping for Genus-0 Surfaces by Harmonic Energy Minimization

Rongjie Lai · Zaiwen Wen · Wotao Yin · Xianfeng Gu · Lok Ming Lui

Received: 18 January 2013 / Revised: 26 June 2013 / Accepted: 3 July 2013 /
Published online: 17 July 2013

Abstract Surface conformal maps between genus-0 surfaces play important roles in applied mathematics and engineering, with applications in medical image analysis and computer graphics. Previous work (Gu and Yau in Commun Inf Syst 2(2):121–146, 2002) introduces a variational approach, where global conformal parameterization of genus-0 surfaces was addressed through minimizing the harmonic energy, with two weaknesses: its gradient descent iteration is slow, and its solutions contain undesired parameterization foldings when the underlying surface has long sharp features. In this paper, we propose an algorithm that significantly accelerates the harmonic energy minimization and a method that iteratively removes foldings by taking advantages of the weighted Laplace–Beltrami eigen-projection. Experimental results show that the proposed approaches compute genus-0 surface harmonic maps much faster than the existing algorithm in Gu and Yau (Commun Inf Syst 2(2):121–146, 2002) and the new results contain no foldings.

Keywords Harmonic energy minimization · Conformal map · Optimization with orthogonality constraints · Weighted Laplace–Beltrami eigenfunctions

R. Lai (✉)
Department of Mathematics, University of Southern California, Los Angeles, CA, USA
e-mail: rongjiel@usc.edu

Z. Wen
Department of Mathematics, MOE-LSC and Institute of Natural Sciences, Shanghai Jiaotong University, Shanghai, China

W. Yin
Department of Computational and Applied Mathematics, Rice University, Houston, TX, USA

X. Gu
Department of Computer Science, State University of New York at Stony Brook, Stony Brook, NY, USA

L. M. Lui
Department of Mathematics, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong

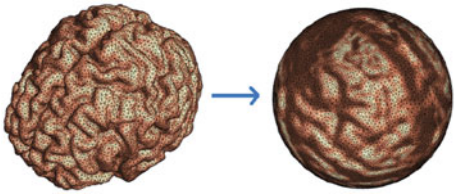

Fig. 1 A conformal parametrization from a brain surface to the unit sphere

1 Introduction

Surface parameterization is the process of mapping a surface onto a simple domain, such as a unit sphere or 2D rectangle. It allows operations on the surface to be carried out on the simple parameter domain. A special type of parameterization is called the conformal parameterization. Under the conformal parameterization, angles and thus local geometry are well preserved. Surface conformal parameterization has been widely used in different areas such as medical image analysis and computer graphics. For example, in medical image analysis, human brains are often conformally mapped to a unit sphere (Fig. 1). Computations and analysis of the brain surface can then be carried on the simple sphere, rather than on the complicated brain cortical surface [2,3]. In computer graphics, surface conformal parameterization is applied for texture mapping and solving PDEs on surfaces [4].

Several local conformal parameterization algorithms have been proposed by different research groups. Levy et al. [5] compute a conformal parameterization of topological disks by approximating the Cauchy–Riemann equation using the least squares method. Eck et al. [6] introduce the discrete harmonic map, which approximates the continuous harmonic map [7] by minimizing a metric dispersion criterion. In [8], Desbrun et al. use conformal maps to define geometry maps, where they compute the conformal maps from a topological disk to the complex plane. Kanai et al. [9] use a harmonic map for geometric metamorphosis. These works mainly deal with the local conformal parameterizations of surface patches, which are homeomorphic to the topological disk.

In many situations, a *global* conformal parameterization that maps a surface onto one global parameter domain is desirable. The global nature avoids the needs of introducing cuts on the surface and partitioning the surface into several patches. Computations can then be performed on one simple parameter domain. Spherical conformal parameterization has been widely used in many different areas such as human brain mapping, computer vision and computer graphics, just to name a few. In this paper, we consider global conformal map from a genus-0 surface to the unit sphere.

Several global conformal parameterization methods have been proposed. Hurdal et al. [10] propose a circle packing approach for spherical conformal parameterization. This method is based on the mean value property of the harmonic map and does not consider the specific metric. Stereographic projection method is also considered for computing conformal parameterization from genus-0 surfaces to the unit sphere. In this method, a face on the mesh is punctured to change the topology of the surface and is then conformally mapped to the 2D plane by either solving a linear system given by the Laplace–Beltrami operator [11] or computing a curvature flow in terms of the conformal factor [12]. After that, stereographic projection is then involved to map the surface onto the sphere. However, artificial chosen of the punctured triangle in this approach will affect the computation results. A more direct approach to face the challenging of the global map is introduced by Gu and Yau [1,13], where a nonlinear optimization method is considered to compute global conformal parameterizations for

genus-0 surfaces. The optimization is done in the tangent spaces of the sphere using gradient descent. This method avoids the stereographic projection and is more stable and accurate. As for the global conformal parameterization of higher-genus surfaces, Gu et al. [1,14] propose to compute the conformal parameterization using the holomorphic 1-form. Curvature flow methods for conformal parameterization of high-genus surfaces, which deform the background Riemannian metric into a uniformization metric, are also proposed by Gu et al. [15,16].

However, there are two weaknesses of the algorithm in [1,13]. First, it often converges slowly due to the gradient descent and the method of projection back to the sphere in their approach. As an example, the conformal parameterization of a surface mesh with 16k vertices takes more than 20 min to compute. This hinders the application of this algorithm. Second, undesired foldings occur in the parameterization of long sharp features of the surface. This leads to inaccurate global parameterizations. The robustness of the algorithm cannot be guaranteed especially for surfaces with complicated geometries. To address these issues, we propose in this paper new methods to compute conformal maps from genus-0 surfaces to the unit sphere.

To compute global spherical *conformal* parameterizations (i.e., conformal map) of genus-0 closed surfaces, we introduce an efficient approach for minimizing the harmonic energy of the map, based on the fact that a *diffeomorphic map* between two genus-0 surfaces $\mathcal{M}$ and S^2 is conformal if and only if it is a local minimizer of the harmonic/Dirichlet energy [17]. It is worth noting that (i) all local minimizers give the same harmonic energy, and (ii) this result does not hold if the map is not a diffeomorphism. The constant map $f(x) \equiv y_0 \in \mathcal{S}^2$, for example, minimizes the harmonic energy to 0 but is clearly not conformal. Hence, the minimization must be confined to the set of diffeomorphisms. In Sect. 3, we introduce an efficient algorithm for minimizing the harmonic energy, but in order to return a local minimum that is a diffeomorphism—a conformal map—the algorithm must start from an initial diffeomorphism. The Gauss map, a common choice of initial map, is generally not a diffeomorphism because on surfaces with long sharp features, it tends to contain *foldings*—parts of the surface being mapped in a way as if they were folded inside other parts. To have a diffeomorphism, we must remove the foldings, and this is a task that no existing maps or algorithms can effectively accomplish without introducing artificial surface cutting. Our attempt is a 3-step heuristic algorithm outlined as follows:

Step 1 Let $\mathbf{F}^0$ be the Gauss map of the input surface.
Step 2 Run an algorithm for minimizing the harmonic energy, which starts from $\mathbf{F}^0$ and returns $\mathbf{F}$. (see Sect. 3)
Step 3 Generate a new map $\mathbf{F}^0$ by applying the weighted Laplace–Beltrami eigen-projection on $\mathbf{F}$ and go to step 2. (see Sect. 4)

We demonstrate that just two or three iterations of Steps 2 and 3 remove the foldings in $\mathbf{F}^0$ and return a folding-free harmonic map $\mathbf{F}$, which is also conformal. Given an initial map $\mathbf{F}^0$ with foldings, the algorithm on Step 2 generally returns a map $\mathbf{F}$ also with foldings, but they appear as localized sharp "singularities." The leading terms of the weighted Laplace–Beltrami eigen-decomposition of $\mathbf{F}$, alike those of the Fourier transform, tend to capture the overall geometry of $\mathbf{F}$ and smooth out the sharps. Consequently, the new initial map $\mathbf{F}^0$ constructed by using only the first three leading eigen-terms tends to contain fewer foldings, or no foldings at all. Then, step 2 is applied again starting from this new initial map and yields a map $\mathbf{F}$ with fewer or no foldings. Although we can not theoretically guarantee the removal of all foldings by this iterative procedure, our numerical experiments on a variety of

different surfaces show that our code always removed all the foldings and returned a (discrete) harmonic map in no more than three iterations.

Moreover, our algorithm performs very fast. Step 2 applies an optimization algorithm modified from the one in [18]. For harmonic energy minimization, our algorithm is designed in a way to take advantages of optimization techniques such as Barzilai–Borwein (using the current and previous gradients to approximate a Newton step) and non-monotone line search (to ensure convergence) to significantly accelerate its convergence. On problems that the algorithm in [1,2] runs in minutes, our algorithm runs in seconds and returns folding-free discrete global harmonic maps. In addition, this fast algorithm is not necessary to restricted on solving harmonic minimization problem. It can be easily adapted to efficiently solve the harmonic energy with landmark-matching problems [3,19] or the p-harmonic minimization problem. Both of these problems are not straightforward to be handled using the stereographic projection method. These potential applications of our fast algorithm will be further explored in our future work.

The rest of this paper is organized as follows. Section 2 reviews the background of conformal maps and its relation to the harmonic energy minimization problem. Section 3 describes the harmonic energy minimization problem and our algorithm for it, and Sect. 4 describes the folding correction based on weighted Laplace–Beltrami eigenfunctions. Numerical experiments and comparisons with existing methods are presented in Sect. 5. The paper is concluded in Sect. 6.

2 Mathematical Background

In this section, we give a brief review of the harmonic energy minimization problem and its generalization of the p-harmonic energy minimization problem [7,17,20].

To introduce the concepts of the harmonic energy minimization problem and harmonic maps between Riemannian surfaces $\mathcal{M}$ and $\mathcal{N}$, we use $\phi_{\mathcal{M}}(x^1, x^2) : \mathbb{R}^2 \to \mathcal{M} \subseteq \mathbb{R}^3$ and $\phi_{\mathcal{N}}(y^1, y^2) : \mathbb{R}^2 \to \mathcal{N} \subseteq \mathbb{R}^3$ as their local coordinates respectively. The inner product of the tangent vectors at each point of the surface can be represented by its first fundamental form. The first fundamental form on $\mathcal{M}$ can be written as $g = \sum_{i,j} g_{ij} dx^i dx^j$, where $g_{ij} = \frac{\partial \phi_{\mathcal{M}}}{\partial x^i} \cdot \frac{\partial \phi_{\mathcal{M}}}{\partial x^j}$ and $i, j = 1, 2$. Similarly, the first fundamental form on $\mathcal{N}$ can be written as $h = \sum_{i,j} h_{ij} dy^i dy^j$ where $h_{ij} = \frac{\partial \phi_{\mathcal{N}}}{\partial y^i} \cdot \frac{\partial \phi_{\mathcal{N}}}{\partial y^j}$ and $i, j = 1, 2$. Given a smooth map $f : \mathcal{M} \to \mathcal{N}$ with local coordinate representation $f(x^1, x^2) = (f_1(x^1, x^2), f_2(x^1, x^2))$, the harmonic energy density of f is:

$$e(f) = \|\mathrm{d}f\|^2 = \sum_{i,j=1,2} g^{ij} \langle f_* \partial_{x^i}, f_* \partial_{x^j} \rangle_h \tag{1}$$

where (g^{ij}) is the inverse of (g_{ij}) and the inner products of *pushforward* vectors $f_* \partial_{x^i}$ and $f_* \partial_{x^j}$ are:

$$\langle f_* \partial_{x^i}, f_* \partial_{x^j} \rangle_h = \left\langle \sum_{m=1}^{2} \frac{\partial f_m}{\partial x^i} \partial_{y_m}, \sum_{n=1}^{2} \frac{\partial f_n}{\partial x^j} \partial_{y_n} \right\rangle_h = \sum_{m,n=1}^{2} h_{mn} \frac{\partial f_m}{\partial x^i} \frac{\partial f_n}{\partial x^j} \tag{2}$$

This formula also defines a new Riemannian metric $f^*(h)(\mathbf{v}_1, \mathbf{v}_2) := \langle f_*(\mathbf{v}_1), f_*(\mathbf{v}_2) \rangle_h$ on $\mathcal{M}$ from a linear extension of the above formula, called the *pullback metric* induced by f and h.

Let's denote the set of all smooth maps from $\mathcal{M}$ to $\mathcal{N}$ by $\mathbb{S}(\mathcal{M},\mathcal{N})$. The harmonic energy minimization problem is defined as follows:

$$\min_{f\in\mathbb{S}(\mathcal{M},\mathcal{N})} \mathcal{E}(f) = \frac{1}{2}\int_{\mathcal{M}} e(f)\mathrm{d}\mathcal{M}. \tag{3}$$

where $\mathcal{E}(f) = \frac{1}{2}\int_{\mathcal{M}} e(f)\mathrm{d}\mathcal{M}$ is called the *Harmonic energy* or *Dirichlet's energy* of f. Critical points of the harmonic energy $\mathcal{E}$ are called *harmonic maps* from $\mathcal{M}$ to $\mathcal{N}$. In particular, if the target manifold $\mathcal{N}$ is $\mathbb{R}^2$, a harmonic map $f = (f_1, f_2)$ from $\mathcal{M}$ to $\mathbb{R}^2$ are given by two harmonic functions on $\mathcal{M}$.

In this paper, we are particularly interested in harmonic maps from a genus-0 surface $(\mathcal{M}, g)$ to the unit sphere (S^2, g_0). In other words, we consider:

$$\begin{aligned} &f : \mathcal{M} \longrightarrow S^2 \hookrightarrow \mathbb{R}^3 \\ &x \longmapsto y \longmapsto \mathbf{F}(x) = (f_1(x), f_2(x), f_3(x)) \quad s.t. \quad \sum_{i=1}^{3} f_i^2(x) = 1 \end{aligned} \tag{4}$$

According (1), the harmonic energy density in this case is given by:

$$\begin{aligned} e(f) &= \sum_{i,j=1,2} g^{ij}\langle f_*\partial_{x^i}, f_*\partial_{x^j}\rangle_{g_0} = \sum_{i,j=1,2} g^{ij}\left\langle \frac{\partial \mathbf{F}}{\partial x^i}, \frac{\partial \mathbf{F}}{\partial x^j}\right\rangle \\ &= \sum_{i,j=1,2} g^{ij}\sum_{\alpha=1}^{3}\frac{\partial f_\alpha}{\partial x^i}\frac{\partial f_\alpha}{\partial x^j} = \sum_{\alpha=1}^{3}\|\nabla_{\mathcal{M}} f_\alpha\|^2 \end{aligned} \tag{5}$$

where $\nabla_{\mathcal{M}} f_\alpha = \sum_{i,j=1}^{2} g^{ij}\frac{\partial f_\alpha}{\partial x^i}\partial_{x^j}$ are gradient of $f_\alpha, \alpha = 1, 2, 3$ on $\mathcal{M}$.

Therefore, computing the harmonic map from $\mathcal{M}$ to S^2 is equivalent to solving the following optimization problem with spherical constraints:

$$\begin{aligned} \min_{\mathbf{F}=(f_1,f_2,f_3)} \quad & \mathcal{E}(\mathbf{F}) = \frac{1}{2}\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f_1\|^2 + \|\nabla_{\mathcal{M}} f_2\|^2 + \|\nabla_{\mathcal{M}} f_3\|^2 \mathrm{d}\mathcal{M} \\ & s.t. \quad \|\mathbf{F}(x)\|^2 = f_1^2(x) + f_2^2(x) + f_3^2(x) = 1, \quad \forall x \in \mathcal{M}. \end{aligned} \tag{6}$$

We say that the map f is **conformal** if

$$f^*(h) = e^{2u} g \tag{7}$$

with a smooth function $u : \mathcal{M} \to \mathbb{R}$ on $\mathcal{M}$. Intuitively, a map is conformal if it preserves the inner product of the tangent vectors up to a scaling factor, called the *conformal factor* e^{2u}. An immediate consequence is that every conformal map preserves angles.

Harmonic maps from a genus-0 surface $\mathcal{M}$ to the unit sphere S^2 is closely related to surface conformal maps, which preserve the inner product of the tangent vectors up to a scaling factor. In fact, for a diffeomorphism f between two genus-0 surfaces $\mathcal{M}$ and S^2, f is conformal if and only if it is a harmonic map [17]. Therefore, computing a conformal map between two genus-0 surfaces is equivalent to computing a harmonic map between them, which can be obtained by finding a critical point of the energy functional $\mathcal{E}$ in (3).

Remark 1 More generally, we can consider about *p-harmonic map* from a n-dimensional manifold $\mathcal{M}$ to the n-dimensional sphere $S^n \subset \mathbb{R}^{n+1}$, which can be computed as the n-dimensional *p-harmonic energy* minimization problem:

$$\min_{\mathbf{F}=(f_1,\ldots,f_{n+1})\in\mathbb{S}(\mathcal{M},S^n)} \quad \mathcal{E}_p(\mathbf{F}) = \frac{1}{p}\int_{\mathcal{M}} \left(\sum_{k=1}^{n+1} \|\nabla_{\mathcal{M}} f_k\|^2\right)^{p/2} \mathrm{d}\mathcal{M}$$
$$s.t. \quad \sum_{k=1}^{n+1} f_k^2(x) = 1, \quad \forall x \in \mathcal{M}. \tag{8}$$

Our algorithm for solving harmonic energy minimization discussed in Sect. 3 can be easily adapted to solve this general problem.

3 Optimization Over Diffeomorphisms Between Genus-0 Surfaces

According to our discussion in Sect. 2, harmonic maps between a genus-0 surface $\mathcal{M}$ and S^2 can be obtained by solving the optimization problem (6) with spherical constraints. Since the equivalence between conformal maps and harmonic maps $\mathcal{M}$ to S^2, solutions to the optimization problem (6) restricted to the set of diffeomorphic maps also provide conformal maps from the given genus-0 surface $\mathcal{M}$ to the unit sphere. Due to its non-convexity, the problem can have multiple local minimizers. This coincides with the fact that conformal maps from surface $\mathcal{M}$ to the unit sphere are non-unique. However, any two different conformal maps of $\mathcal{M}$ only differ by a Mobius transformation of the unit sphere. In addition, the harmonic energies of all conformal maps are identical [20]. Thus, all diffeomorphic local minimizers of problem (6) have the same harmonic energy, and any one of them gives a harmonic map (or conformal map) from $\mathcal{M}$ to the unit sphere.

However, since it is difficult to characterize diffeomorphic maps at least numerically, our first stage is to solve problem (6) itself but relax the diffeomorphic constraints. Empirical evidence show that the quality of the returned map is usually very good except that the map can have foldings which appear as localized sharp "singularities." We should point out that foldings in our problem are due to the folding in the initial solutions, which are not diffeomorphic. Any existing harmonic-energy-minimization algorithms including ours, if starting from a map with folding, may return a map with foldings. If the initial map has no foldings, our experience shows that the solution has no folding, too. Generally speaking, it is not straightforward to have a diffeomorphism from an arbitrary given genus-0 surface to the unit sphere without introducing artificial cutting. Hence, we first describe an algorithm for solving problem (6) and the next section is devoted to generating a folding-free initial map.

Unlike optimization in $\mathbb{R}^n$ where it is straightforward to decrease the objective along a straight search line (e.g., along the negative gradient direction), it is not as easy to do so in a curved manifold. A natural choice is the geodesic, which is the analog of straight line and has the shortest length between two different points. Another choice is iterative projection: descent along straight lines and project points back to the manifold. There are various optimization methods for optimization on manifold such as [21–23] and references therein, which are mostly based on either geodesics or projections. Considering the fact that at each $x \in \mathcal{M}$, $\|\mathbf{F}(x)\| = 1$ defines a unit sphere, we choose to develop a sphere-geodesic descent method which is essentially identical to the recent work [18]. It is numerically efficient and lets us apply state-of-the-art acceleration techniques such as Barzilai–Borwein steps and non-monotone line search with global convergence guarantees.

3.1 Constraint Preserving Update

The constraint of problem (6) is often referred to as the unit-sphere manifold, which is a special case of the Stiefel manifold. Our constraint preserving scheme can be viewed as a kind of projected gradient method on the manifold. The Lagrangian of problem (6) is

$$\mathcal{L}(\mathbf{F}, \lambda) := \mathcal{E}(\mathbf{F}) - \frac{1}{2}\int_{\mathcal{M}} \lambda(x)\left(\|\mathbf{F}(x)\|^2 - 1\right) \mathrm{d}\mathcal{M},$$

where λ is the Lagrange multiplier. The first-order optimality conditions of (6) are (assuming they are well-defined)

$$\begin{cases} \operatorname{grad}_{\mathbf{F}} \mathcal{L}(\mathbf{F}, \lambda) := H - \lambda \mathbf{F} = 0, \\ \quad \|\mathbf{F}(x)\| = 1, \quad \forall x \in \mathcal{M}, \end{cases} \tag{9}$$

where $H = \operatorname{grad} \mathcal{E}(\mathbf{F}) = -(\triangle_{\mathcal{M}} f_1, \triangle_{\mathcal{M}} f_2, \triangle_{\mathcal{M}} f_3)$ is the Fréchet derivative of $\mathcal{E}(\mathbf{F})$ with respect to $\mathbf{F}$ and $\triangle_{\mathcal{M}}$ is the Laplace–Beltrami operator of $\mathcal{M}$. For concise notation, we let A^*F denote the function $(A^*F)(x) := \langle A(x), \mathbf{F}(x)\rangle$. Applying the linear operator $\mathbf{F}^*$ to both sides of the first equation in (9) and using the fact $(\mathbf{F}^*\mathbf{F})(x) = \|\mathbf{F}(x)\| = 1,\ \forall x \in \mathcal{M}$, we obtain $\lambda = \mathbf{F}^*H = H^*\mathbf{F}$. Plugging λ back to (9) gives $0 = H - (\mathbf{F}^*H)\mathbf{F} = H(\mathbf{F}^*\mathbf{F}) - \mathbf{F}(H^*\mathbf{F})$ or, equivalently

$$A\mathbf{F} = 0 \ \text{ with } \ A := H\mathbf{F}^* - \mathbf{F}H^*.$$

By definition, $A(x)$ is skew-symmetric at every $x \in \mathcal{M}$. Following [18], we use A and its skew-symmetry to define a search path maintaining $\|\mathbf{F}\| = 1$.

Observe that $A(x)\mathbf{F}(x)$ is the gradient of $\mathcal{E}$ at x projected to S^2. In $\mathbb{R}^3$, the steepest descent path is $\mathbf{Y}(x) := \mathbf{F}(x) - \tau A(x)\mathbf{F}(x)$, where τ is a scalar representing the step size. However, this $\mathbf{Y}(x)$ does not generally have a unit norm. If we instead apply the implicit update

$$\mathbf{Y}(x) = \mathbf{F}(x) - \frac{\tau}{2}A(x)(\mathbf{F}(x) + \mathbf{Y}(x))$$

and obtain

$$\mathbf{Y}(x) = \left(I + \frac{\tau}{2}A(x)\right)^{-1}\left(I - \frac{\tau}{2}A(x)\right)\mathbf{F}(x), \tag{10}$$

then the fact that $\left(I + \frac{\tau}{2}A(x)\right)^{-1}\left(I - \frac{\tau}{2}A(x)\right)$ is orthogonal gives us $\|\mathbf{Y}(x)\| = \|\mathbf{F}(x)\| = 1$. Hence, we define the update path $\mathbf{Y}[\tau]$ by

$$\mathbf{Y}[\tau] := \mathbf{F} - \frac{\tau}{2}A(\mathbf{F} + \mathbf{Y}[\tau]). \tag{11}$$

Theorem 1 below shows that the constraints are preserved at every τ. Since the Eq. (11) is linear with respect to $\mathbf{Y}[\tau]$, its closed-form solution can be computed explicitly as a linear combination of $\mathbf{F}$ and H, in which the linear coefficients are determined by τ, $\|\mathbf{F}\|$, $\|H\|$ and $\mathbf{F}^*H$.

Theorem 1 *For every τ, $Y[\tau]$ of (11) satisfies $\|\mathbf{Y}[\tau]\| = \|\mathbf{F}\|$ point-wise. In addition, it is given in the closed-form*

$$\mathbf{Y}[\tau] = \left(I + \frac{\tau}{2}A\right)^{-1}\left(I - \frac{\tau}{2}A\right)\mathbf{F},$$

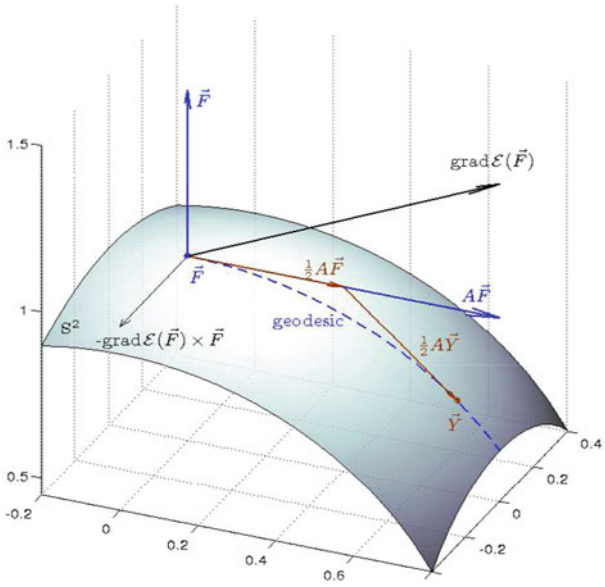

Fig. 2 An illustration of constraints preserving update. Given x, $\|\mathbf{F}(x)\| = 1$. The point $\mathbf{F}(x) - \tau A(x)\mathbf{F}(x)$ is not feasible, but $\mathbf{Y}(x)$ given by (10) is feasible

which can be computed as $\mathbf{Y}[\tau] = \alpha[\tau]\mathbf{F} + \beta[\tau]H$, *Here,*

$$\alpha[\tau] = \frac{\left(1 + \frac{\tau}{2}\mathbf{F}^*H\right)^2 - \left(\frac{\tau}{2}\right)^2 \|\mathbf{F}\|^2\|H\|^2}{1 - \left(\frac{\tau}{2}\right)^2 (\mathbf{F}^*H)^2 + \left(\frac{\tau}{2}\right)^2 \|\mathbf{F}\|^2\|H\|^2},$$

$$\beta[\tau] = \frac{-\tau\|\mathbf{F}\|^2}{1 - \left(\frac{\tau}{2}\right)^2 (\mathbf{F}^*H)^2 + \left(\frac{\tau}{2}\right)^2 \|\mathbf{F}\|^2\|H\|^2}.$$

We refer [18] for the details of the proof of this theorem. The result of theorem is visualized in Fig. 2.

According to Theorem 1, the cost of computing $\mathbf{Y}[\tau]$ is dominated by the computation of $\|H\|^2$ and $\mathbf{F}^*H$. It is also worth noting that τ sometimes needs updates, which incurs updates to $\mathbf{Y}[\tau]$, but the cost is relatively small.

3.2 Algorithm and Initial Map

To make the maximal use of the computed $\|H\|^2$ and $\mathbf{F}^*H$ at each iteration k, we determine a step size τ_k that makes significant descent while still guarantees the convergence of the overall iterations. To this end, instead of the classical Armijo–Wolfe based monotone line search, we apply nonmonotone curvilinear[1] search with an initial step size determined by the Barzilai–Borwein formula, which we have found more efficient for our problem. They were developed originally for $\mathbb{R}^n$ in [24] and [25], respectively. At iteration k, the step size is computed as

$$\tau_{k,1} = \frac{\int_{\mathcal{M}} \|D_{k-1}(x)\|^2 \mathrm{d}\mathcal{M}}{|\int_{\mathcal{M}} D^*_{k-1}(x) W_{k-1}(x)\mathrm{d}\mathcal{M}|} \quad \text{or} \quad \tau_{k,2} = \frac{|\int_{\mathcal{M}} D^*_{k-1}(x) W_{k-1}(x)\mathrm{d}\mathcal{M}|}{\int_{\mathcal{M}} \|W_{k-1}(x)\|^2 \mathrm{d}\mathcal{M}}, \quad (12)$$

[1] As our search path is a curve rather than a straight line.

where $D_{k-1} := \mathbf{F}_k - \mathbf{F}_{k-1}$ and $W_{k-1} = A_k\mathbf{F}_k - A_{k-1}\mathbf{F}_{k-1}$. The final value for τ_k is a fraction (up to 1, inclusive) of $\tau_{k,1}$ or $\tau_{k,2}$ determined by the nonmonotone search in Algorithm 1, Lines 3 and 5, which enforce a trend of descent in the objective value but do not require strict descent at each iteration. At the first iteration where $\mathbf{F}_{k-1}$ and A_{k-1} are not available, one can set a unit initial step size. The convergence of this algorithm can be obtained by extending the proof in [18] ofr $\mathbb{R}^n$ in our setting.

Assembling the above parts, we arrive at Algorithm 1, in which ϵ is a stopping parameter, and ρ, δ, and ξ are curvilinear search parameters, which can be set to typical values as 10^{-4}, 0.1 and 0.85, respectively.

Algorithm 1: A Fast Algorithm

1 Given $\mathbf{F}_0$, pick $\rho,\ \delta,\ \xi,\ \epsilon \in (0, 1)$. $k \leftarrow 0$.
2 **while** $\|\nabla\mathcal{E}(\mathbf{F}_k)\| > \epsilon$ **do**
3 Compute $\tau_k \leftarrow \tau_{k,1}\delta^h$ or $\tau_k \leftarrow \tau_{k,2}\delta^h$, where h is the smallest nonnegative integer satisfying $\mathcal{E}(\mathbf{Y}_k(\tau_k)) \le C_k + \rho\tau_k\mathcal{E}'(\mathbf{Y}_k(0))$.
4 $\mathbf{F}_{k+1} \leftarrow \mathbf{Y}_k(\tau_k)$.
5 $Q_{k+1} \leftarrow \xi Q_k + 1$ and $C_{k+1} \leftarrow \frac{\xi Q_k C_k + \mathcal{E}(\mathbf{F}_{k+1})}{Q_{k+1}}$.
6 $k \leftarrow k + 1$.

Generally speaking, it is difficult to construct one-to-one and onto smooth maps from a given genus-0 surface to the unit sphere. In practice, we choose the Gauss map as the initial map $\mathbf{F}_0$, which is defined as follows:

Definition 1 (*Gauss map*) $\mathcal{G} : \mathcal{M} \to S^2$, $\mathcal{G}(p) = \mathbf{n}_p$, where $\mathbf{n}_p$ is the unit normal vector at $p \in \mathcal{M}$.

As we shall see later, when Gauss map contains foldings—which are common for surfaces with long and sharp parts—the initial map is not a diffeomorphism. It leads to foldings in the solutions of our discrete algorithm below. We address foldings in subsection 3.4 and their removal in Sect. 4.

3.3 Implementation and Simulations

In the implementation of Algorithm 1, we approximate $\mathcal{M}$ by a triangulated surface $\mathcal{M} = \{V = \{p_i\}_{i=1}^N, T = \{T_l\}_{l=1}^L\}$, where $p_i \in \mathbb{R}^3$ is the i-th vertex and T_l is the l-th triangle. For a function $h = (h(p_1), \ldots, h(p_N))$ defined on the triangle mesh, we approximate the Laplace–Beltrami operator and numerical integral on surface $\mathcal{M}$ by [26–28]:

$$\triangle_{\mathcal{M}} h(p_i) \approx \frac{3}{\sum_{p_i \in T_l} \text{Area}(T_l)} \sum_{j \in N_i} \omega_{ij}(p_i)\left(h(p_j) - h(p_i)\right),$$

$$\int_{\mathcal{M}} h \mathrm{d}_{\mathcal{M}}(x) \approx \sum_{i=1}^{N} h(p_i) \cdot A_i \tag{13}$$

where $\omega_{ij}(p_i) = \frac{cot\alpha_{ij}(p_i) + cot\beta_{ij}(p_i)}{2}$, α_{ij} and β_{ij} are the two angles opposite to the edge $\overline{p_i p_j}$, N_i is the first ring neighborhood of the vertex p_i, and $A_i = \frac{1}{3}\sum_{p_i \in T_l} \text{Area}(T_l)$.

For a general given genus-0 surface $(\mathcal{M}, g)$, there is no analytical form of the harmonic maps from $(\mathcal{M}, g)$ to (S^2, g_0). However, a harmonic map $\mathbf{F}$ from $\mathcal{M}$ to S^2 is also conformal.

Table 1 Iterations and computation times for surfaces in Fig. 3

Surface	# of vertices	# of iterations	Time (s)	ϵ
Putamen	10,000	1, 414	4.80	1e−10
Brain	15,002	1, 329	9.36	1e−10
Maxplanck	12,556	892	5.19	1e−10

Namely, the harmonic map $\mathbf{F}$ will preserve angles of tangent vectors and satisfying $\mathbf{F}^*(g_0) = e^{2u}g$. Here, we would like to check the accuracy of the resulting harmonic map $\mathbf{F}$ by checking its conformality. Given a surface map $\mathbf{F} : \mathcal{M} \to S^2, p \mapsto (f_1(p), f_2(p), f_3(p))$, the "conformal factor" with respect to the map $\mathbf{F}$ can be approximated by:

$$e^{2u(p_i)} = \frac{\sum_{p_i \in T_l} \text{Area}(\mathbf{F}(T_l))}{\sum_{p_i \in T_l} \text{Area}(T_l)} \tag{14}$$

Table 1 and Fig. 3 give the results of Algorithm 1 on three different examples: a Putamen surface, a brain surface and a Maxplanck surface. Their harmonic maps are obtained by Algorithm 1 with a fixed $\epsilon = 10^{-10}$ and initialized by the Gauss map. The surface sizes, numbers of iterations and computation times are given in Table 1. The top row of Fig. 4 shows how the energies decrease over the iterations. To illustrate the quality of the result maps, we compute the angle differences between triangles on the input surfaces and the corresponding triangles on the obtained maps. As shown in the histograms on the bottom row of Fig. 4, most of the angle differences are close to zero, so the obtained maps do preserve angles and thus nearly satisfy the main property of the harmonic map. In general, Algorithm 1 can efficiently compute harmonic maps of a large class of surfaces as long as they do not contain extremely long and sharp patterns.

3.4 Artificial Folding Issue

For arbitrary given genus-0 surface $\mathcal{M}$, it is not easy to obtain an initial diffeomorphism from $\mathcal{M}$ to the unit sphere S^2. The most natural map from $\mathcal{M}$ to S^2 is the Gauss map. However, the initial Gauss map can introduce artificial foldings for surfaces with complicated geometries, especially for those with long sharp features. The Gauss map as the initial map with foldings often cause singularities in the solution of algorithm for harmonic energy minimization. Such phenomenon has also been observed in the well-known algorithm [1,13], where artificial cutting was introduced to overcome this issue [13].

Figure 5 depicts the singularities in the solution near the long neck, tail and four legs of a dinosaur surface. To tackle this difficulty, we introduce a folding removal method based on the weighted Laplace–Beltrami eigen-projection in the next section.

4 Folding Removal by Weighted Laplace–Beltrami Eigenfunctions

Since the possible foldings in the maps out of our Algorithm 1 are localized and sharp, we novelly propose to "smooth" them out using the leading three terms of the weighted Laplace–Beltrami (LB) eigenfunctions on $\mathcal{M}$. To clearly introduce our folding removal method, we first present our simple observation on spherical harmonics and then extend it to weighted LB eigenfunctions for the folding removal algorithm.

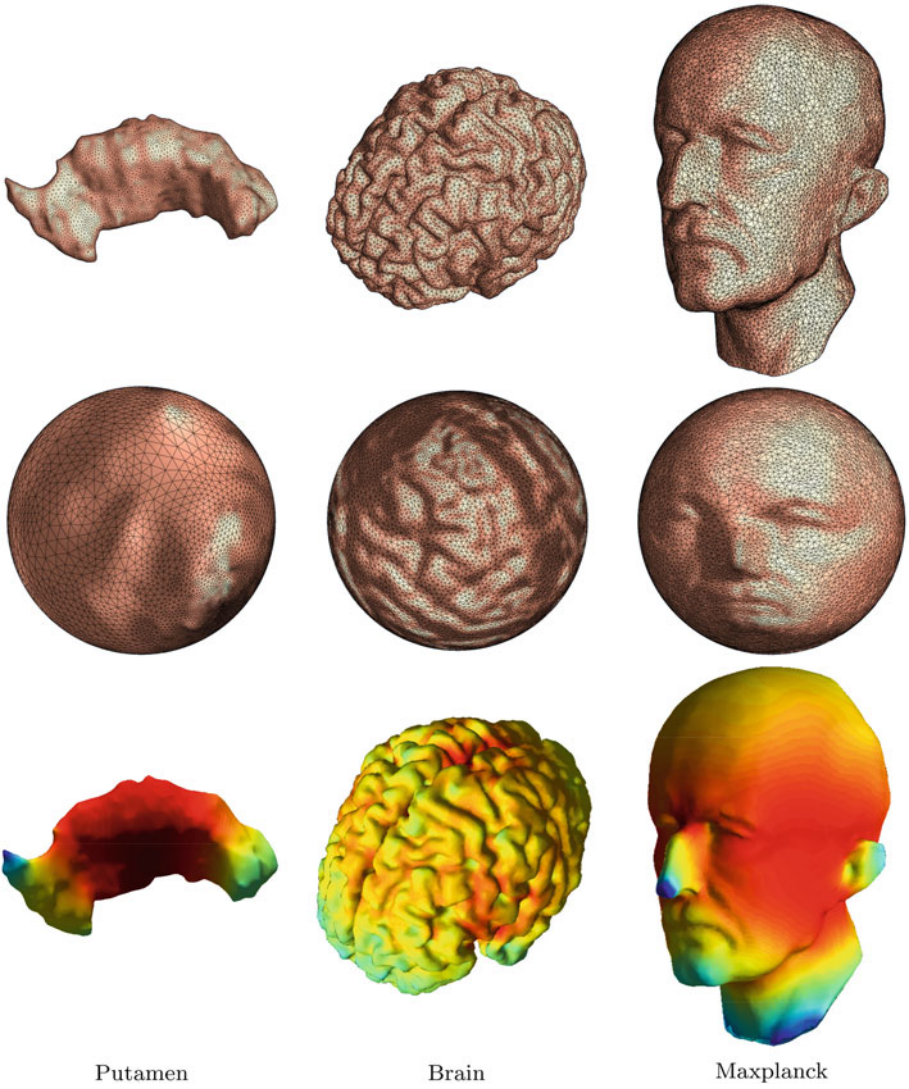

Fig. 3 *First row* three input surfaces; *second row* harmonic maps obtained by Algorithm 1; *third row* surfaces *color*-coded by the corresponding u in the conformal factors (Color figure online)

4.1 Laplace–Beltrami Eigenfunctions and Spherical Harmonics

Given a closed Riemannian surface $(\mathcal{M}, g)$, its Laplace–Beltrami (LB) operator is defined as [20,29]:

$$\triangle_g \phi = \frac{1}{\sqrt{G}} \sum_{i=1}^{2} \frac{\partial}{\partial x_i} \left(\sqrt{G} \sum_{j=1}^{2} g^{ij} \frac{\partial \phi}{\partial x_j} \right) \tag{15}$$

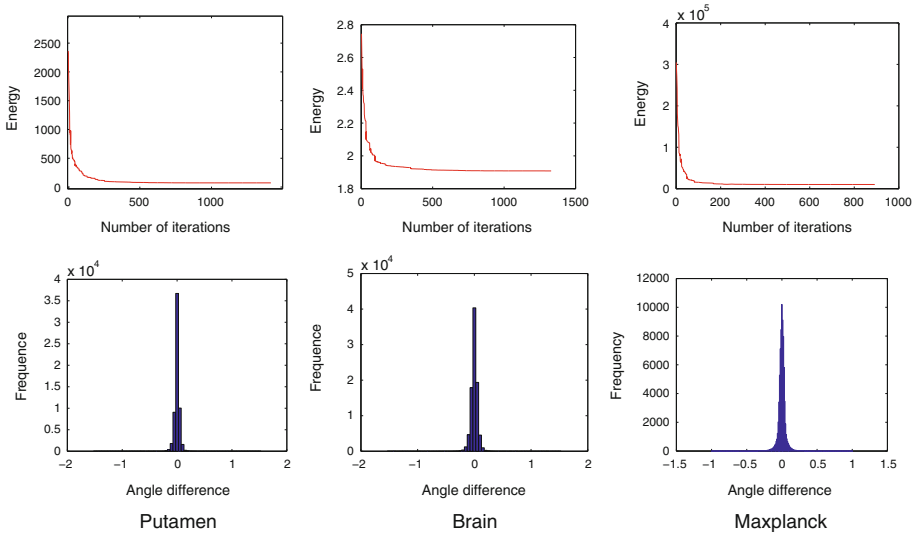

Fig. 4 *Top row* harmonic energy (6) versus iteration number; *bottom row* angle difference histograms

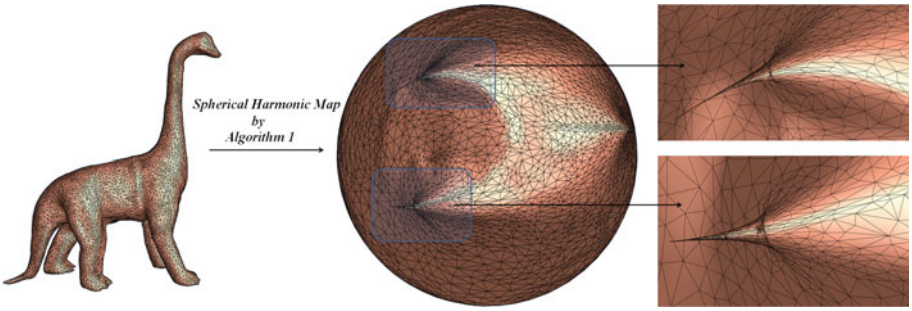

Fig. 5 The artificial foldings in the dinosaur surface

where (g^{ij}) is the inverse matrix of $g = (g_{ij})$ and $G = \det(g_{ij})$. The LB operator is self adjoint and elliptic, so its spectrum is discrete. We let the eigenvalues of $\triangle_g$ be denoted as $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots$ and the corresponding eigenfunctions as $\phi_0, \phi_1, \phi_2, \ldots$ such that

$$\triangle_g \phi_n = -\lambda_n \phi_n, \quad \int_{\mathcal{M}} \phi_n^2 \mathrm{d}\mathcal{M} = 1, \quad n = 0, 1, 2, \ldots. \tag{16}$$

Then $\{\phi_n \mid n = 0, 1, 2, \ldots\}$ forms an orthonormal basis of the smooth function space on $\mathcal{M}$. A well-known example of the LB eigen-problems is the LB eigen-problem of the unit sphere (S^2, g_0), namely, $\triangle_{g_0} \phi_n = -\lambda_n \phi_n$, $n = 0, 1, 2, \ldots$, whose solutions are spherical harmonic functions. More specifically, if we use the standard spherical coordinate (θ, ξ) for the unit sphere in $\mathbb{R}^3$:

$$\begin{cases} x = \sin\theta \cos\xi, \\ y = \sin\theta \sin\xi, \\ z = \cos\theta, \end{cases} \qquad \theta \in [0, \pi], \ \xi \in [0, 2\pi), \tag{17}$$

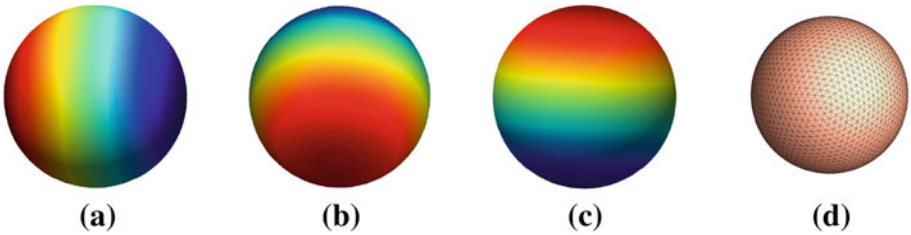

Fig. 6 **a–c** The first three nontrivial LB eigenfunctions ϕ_1, ϕ_2 and ϕ_3 *color*-coded on the unit sphere. **d** The surface reconstructed by ϕ_1, ϕ_2 and ϕ_3 (Color figure online)

the spherical harmonic functions can be written as smooth functions in θ and ξ. We are especially interested in the first three nontrivial spherical harmonic functions:

$$\begin{cases} \phi_1 = \frac{1}{2}\sqrt{\frac{3}{\pi}}\sin\theta\cos\xi, \\ \phi_2 = \frac{1}{2}\sqrt{\frac{3}{\pi}}\sin\theta\sin\xi, \\ \phi_3 = \frac{1}{2}\sqrt{\frac{3}{\pi}}\cos\theta. \end{cases} \qquad \theta \in [0,\pi],\ \xi \in [0,2\pi), \tag{18}$$

An interesting observation is that the first three nontrivial spherical harmonic functions ϕ_1, ϕ_2, ϕ_3 provide us with left-right, up-down and forward-backward structures of the given sphere (see Fig. 6). More precisely, since $\phi_1^2 + \phi_2^2 + \phi_3^2 = \frac{3}{4\pi}$, one can use $\mathbf{\Phi} = (\phi_1, \phi_2, \phi_3)$ to construct a diffeomorphism as:

$$\mathfrak{S}_{\mathbf{\Phi}} : S^2 \to S^2, \quad p \mapsto \mathfrak{S}_{\mathbf{\Phi}(p)} = \frac{\mathbf{\Phi}(p) - \mathbf{c}}{\|\mathbf{\Phi}(p) - \mathbf{c}\|}, \tag{19}$$

where $\mathbf{c} = \frac{\int_{S^2} \mathbf{\Phi} \mathrm{d}S^2}{\int_{S^2} \mathrm{d}S^2} = \mathbf{0}$ and $\mathfrak{S}_{\mathbf{\Phi}}$ is called the *star map* defined from $\mathbf{\Phi} = (\phi_1, \phi_2, \phi_3)$.

An advantage of using $\mathbf{\Phi} = \{\phi_1, \phi_2, \phi_3\}$ to construct the Star map is that $\{\phi_1, \phi_2, \phi_3\}$ are all intrinsically defined on the unit sphere. Thus the Star map construction in this way does not depend on the chosen of Euclidean coordinate representation of the sphere. Moreover, due to the analytic form of ϕ_1, ϕ_2, ϕ_3 in (18), the Star map obtained from $\mathbf{\Phi}$ is a smooth one-to-one and onto map from S^2 to S^2. We would like to generalize this observation to obtain a one-to-one and onto map from arbitrary given genus-0 surface $(\mathcal{M}, g)$ to the unit sphere, which is described in the following subsection.

4.2 Weighed LB Eigenfunctions and Folding Removal Algorithm

A general given genus-0 closed surface $(\mathcal{M}, g)$ can be viewed as a topological manifold S^2 with a given metric g. Since there always exits a conformal map between $\mathcal{M} = (S^2, g)$ and (S^2, g_0) [7], we write the corresponding conformal factor of (S^2, g) as e^{2u}. Namely, we have $e^{2u} g = g_0$.

Theorem 2 *Given a genus-0 surface $(\mathcal{M}, g)$ with conformal factor e^{2u}. Let $\mathbf{\Phi}^u = (\phi_1^u, \phi_2^u, \phi_3^u)$ be the first three nontrivial eigenfunctions of the weighted Laplace–Beltrami eigensystem:*

$$\triangle_g \phi_n^u = -\lambda e^{2u} \phi_n^u, \quad n = 0, 1, 2, \ldots \tag{20}$$

Write $\mathbf{c} = \frac{\int_{\mathcal{M}} \mathbf{\Phi}^u \mathrm{d}\mathcal{M}}{\int_{\mathcal{M}} \mathrm{d}\mathcal{M}}$. *Then the Star map:*

$$\mathfrak{S}_{\mathbf{\Phi}^u} : \mathcal{M} \to S^2, \quad \mathfrak{S}_{\mathbf{\Phi}^u}(p) = \frac{\mathbf{\Phi}^u(p) - \mathbf{c}}{\|\mathbf{\Phi}^u(p) - \mathbf{c}\|},$$

defined from Φ^u *is a smooth one-to-one and onto map from* $\mathcal{M}$ *to* S^2.

Proof Since $\mathcal{M}$ is a genus-0 surface, the underlying topological manifold of (M, g) is a sphere. Consider the Laplace–Beltrami eigen-problems on $\mathcal{M}$ with respect to the standard metric of the unit sphere g_0:

$$\triangle_{g_0}\phi_n = -\lambda_n\phi_n, \ n = 0, 1, 2, \ldots. \tag{21}$$

Therefore, the solutions of (21) on $\mathcal{M}$ are spherical harmonic functions. On the other hand, since $g_0 = e^{2u}g$, the definition of the Laplace–Beltrami operator in (15) provides:

$$\triangle_{e^{2u}g}\phi = \frac{1}{e^{2u}\sqrt{G}} \sum_{i=1}^{2} \frac{\partial}{\partial x_i} \left(e^{2u}\sqrt{G} \sum_{j=1}^{2} e^{-2u} g^{ij} \frac{\partial \phi}{\partial x_j} \right) = e^{-2u}\triangle_g \phi, \tag{22}$$

where (g^{ij}) is the inverse matrix of $g = (g_{ij})$ and $G = \det(g_{ij})$. By combining with (21) and (22), we have the solutions of weighted Laplace–Beltrami eigensystem:

$$\triangle_g \phi_n^u = -e^{2u}\lambda_n\phi_n^u, \quad n = 0, 1, 2, \ldots \tag{23}$$

are spherical harmonic functions. Therefore, from the analytic expression of the first three spherical harmonic functions in (18), $\mathbf{\Phi}^u = (\phi_1^u, \phi_2^u, \phi_3^u)$ should satisfy $(\phi_1^u)^2 + (\phi_2^u)^2 + (\phi_3^u)^2 = \frac{3}{4\pi}$. More importantly, $\mathfrak{S}_{\mathbf{\Phi}}^u$ is a diffeomorphism from $\mathcal{M}$ to the unit sphere S^2. □

The above theorem provides a constructive approach to have a one-to-one and onto diffeomorphism from an arbitrary given genus-0 surface $(\mathcal{M}, g)$ to the unit sphere. Then a folding free conformal map can be obtained by using this map as an initial map in the Algorithm 1. The conformal factor e^{2u} of $\mathcal{M}$ in the above weighted Lapalce-Beltrami eigensystem is crucial. It is not true that the first three nontrivial eigenfunctions standard LB eigensystem without weight will provide the diffeomorphism. We would like to point out that the conformal factor e^{2u} of $(\mathcal{M}, g)$ is related to a highly nonlinear equation

$$\triangle_g u + K - e^{2u} = 0, \qquad \text{where } K \text{ is the Guass curvature of } \mathcal{M} \tag{24}$$

whose solution is not straightforward to have [7,12,30].

As we discussed in Sect. 3.4, for a surface $\mathcal{M}$ with long and sharp parts like the dinosaur neck, tail and legs in Fig. 5, the result mapping $\mathbf{F}_0 : \mathcal{M} \to S^2$ from Algorithm 1 contains artificial foldings introduced by the initial Gauss map. More importantly, those artificial foldings appear as localized sharp "singularities". In this case, we can view $\mathbf{F}_0$ as an approximation of the conformal map which can give us the first approximation of the "conformal factor", e^{2u_0}, with respect to $\mathbf{F}_0$. However, e^{2u_0} might be inaccurate in these localized singularites. Due to the global properties of the leading terms of the weighted LB eigen-system, alike those of the Fourier transform, the corresponding Star map $\mathfrak{S}_{\mathbf{\Phi}^u}$ tends to capture the overall geometry and smooth out these sharps. This motives us to propose the following Algorithm 2 for computing a folding free harmonic map with folding removal iteration using weighted LB eigen-projection, which is also illustrated in Fig. 7. We also would like to point out that this algorithm is currently lack of rigorous proof, while it does provide good conformal maps

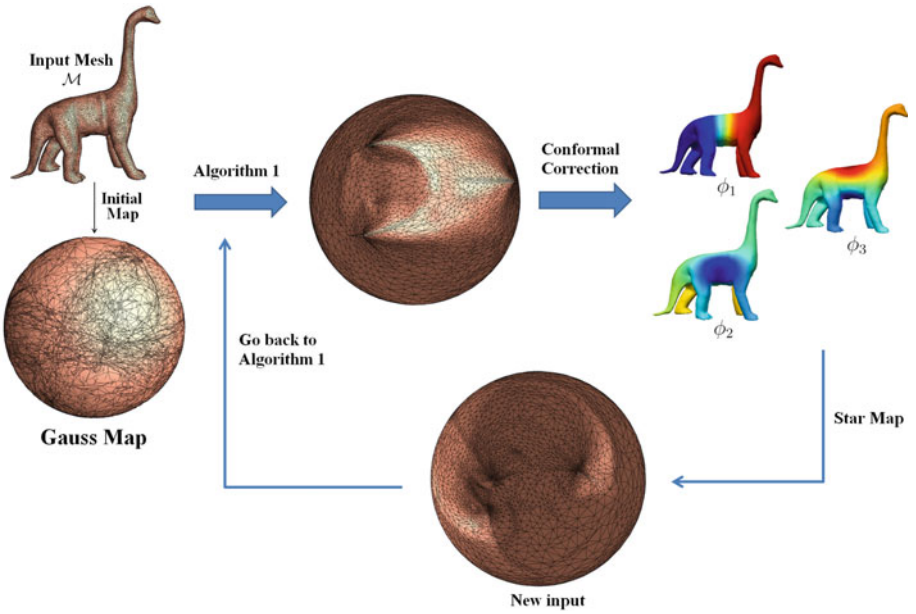

Fig. 7 An illustration of Algorithm 2

for surfaces with long and sharp features in practice. A rigorous mathematical validation of this algorithm will be certainly considered in our future work.

As we can see from the Fig. 9, the conformal map before the folding removal iteration has foldings, and the obtained conformal factor is not accurate enough. Thus, the visualization of $\mathbf{\Phi}^u = (\phi_1^u, \phi_2^u, \phi_3^u)$ is not a sphere but an ellipsoid. While, the start map from this ellipsoid will certainly provide a good initial map. From the two steps of the proposed folding removal iteration illustrated in last two rows of Fig. 9, we can see that the resulting conformal map is folding-free, and the corresponding conformal factor is accurate. Therefore the $\mathbf{\Phi}^u = (\phi_1^u, \phi_u^2, \phi_3^u)$ is a sphere as we state in the Theorem 2.

Algorithm 2: Folding removal by weighted LB Eigen-projection

1. Compute a map $\mathbf{F}_0 : \mathcal{M} \to S^2$ using Algorithm 1.
   Compute the corresponding e^{2u_0} using the approximation formula (14).

Iterate the following steps starting from $k = 1$.

2. Given the map $\mathbf{F}_{k-1}$ and conformal factor $e^{2u_{k-1}}$, solve (20).
3. Construct a Star map using $\{\phi_1^{u_{k-1}}, \phi_2^{u_{k-1}}, \phi_3^{u_{k-1}}\}$.
4. Start Algorithm (1) for the Star map and obtain $\mathbf{F}_k$ and u_k.

Similar to using the finite element method (FEM) for computing the LB eigen-system on triangulated surfaces [31–34], we use FEM to solve the above weighted LB eigen-system (23). For any given surface $\mathcal{M}$ in $\mathbb{R}^3$, we represent $\mathcal{M}$ as a triangular mesh $\left\{V = \{p_i\}_{i=1}^N, T = \{T_l\}_{l=1}^L\right\}$, where $p_i \in \mathbb{R}^3$ is the ith vertex and T_l is the l-th triangle. One can choose linear elements $\{e_i\}_{i=1}^N$, which satisfy $e_i(p_j) = \delta_{i,j}$ in the Kronecker delta notion, and write $\mathbb{E} = Span_{\mathbb{R}}\{e_i\}_{i=1}^N$. Then the discrete version of the weak form of the continuous problem is to find a $\phi \in \mathbb{E}$ such that

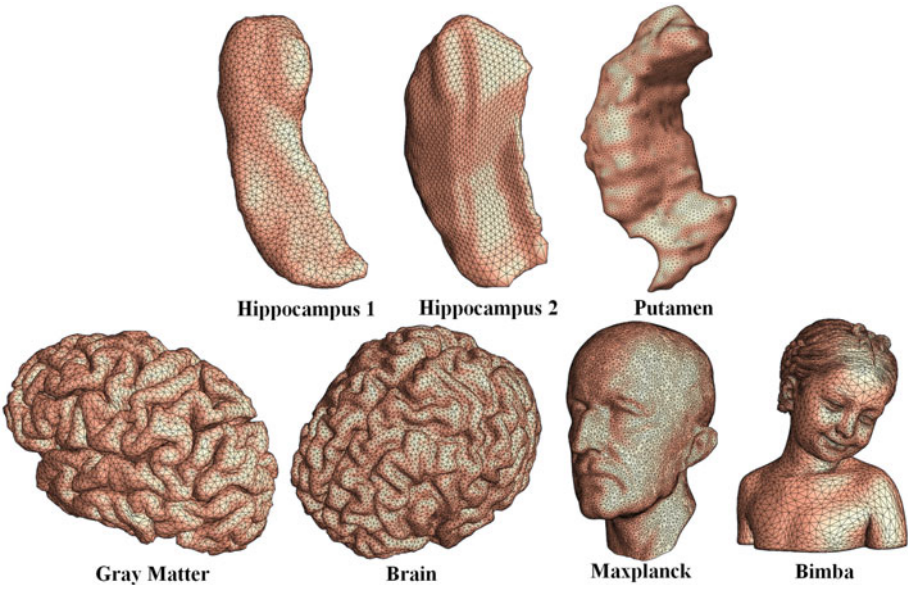

Fig. 8 Surfaces for comparisons listed in Table 2

Table 2 Comparison between the proposed Algorithm 1 and the algorithm in [1]

Surface	# of vertices	ϵ	The proposed Algorithm 1		Algorithm in [1]	
			# of iterations	Time (s)	# of iterations	Time (s)
Hippocampus 1	2,000	1e−10	210	0.19	790	16.91
Hippocampus 2	2,562	1e−10	4,225	5.83	1,594	108.36
Putamen	10,002	1e−10	1,414	4.80	2,684	760.33
Gray matter	10,000	1e−10	1,598	5.79	2,794	822.90
Brain	15,002	1e−10	1,329	9.36	1,560	664.97
Maxplanck	12,556	1e−10	892	5.19	1,990	704.01
Bimba	15,002	1e−10	1,432	17.40	3,197	1,358.72

$$\sum_l \int_{T_l} \nabla_{\mathcal{M}}\phi \nabla_{\mathcal{M}}\eta = \lambda \sum_l \int_{T_l} e^{2u}\phi\eta, \quad \forall \eta \in \mathbb{E}. \tag{25}$$

If we write

$$\begin{cases} \phi = \sum_{i=1}^N x_i e_i \\ A = (a_{ij})_{N\times N}, \quad a_{ij} = \sum_l \int_{T_l} \nabla_{\mathcal{M}} e_i \nabla_{\mathcal{M}} e_j \\ B_u = (b_{ij})_{N\times N}, \quad b_{ij} = \sum_l \int_{T_l} e^{2u} e_i e_j, \end{cases} \tag{26}$$

where the stiffness matrix A is symmetric and the mass matrix B_u is symmetric and positive definite, and the discrete variational problem is equivalent to the generalized matrix eigen-problem:

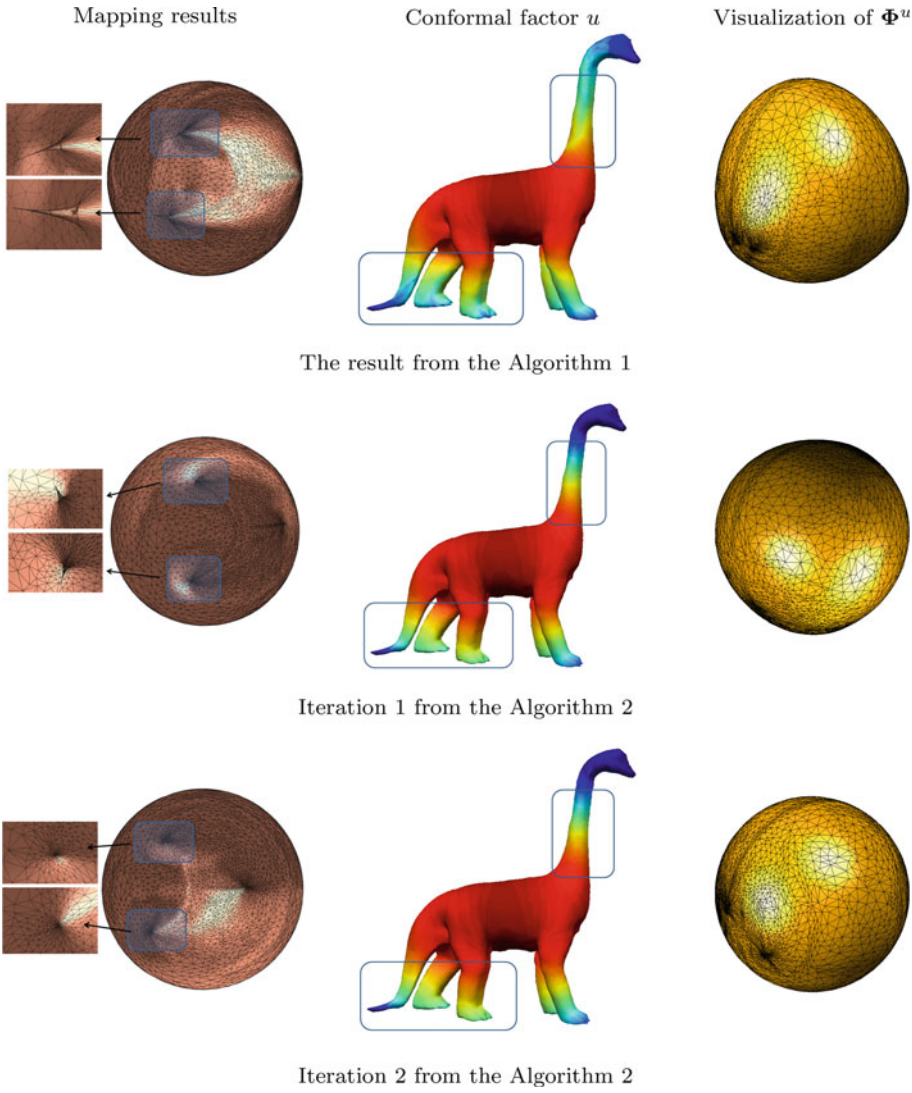

Fig. 9 The *first row* the mapping results, the corresponding conformal factor u and visualization of Φ^u before folding removal iteration; the *second* and *third rows* the mapping results, the corresponding conformal factors and visualization of $\mathbf{\Phi}^u = (\phi_1^u, \phi_2^u, \phi_3^u)$ obtained by one and two steps of folding removal iteration, respectively

$$\begin{cases} Ax = \lambda B_u x, \text{ where } x = (x_1, \ldots, x_N)^T \\ \phi = \sum_{i=1}^{N} x_i e_i. \end{cases} \tag{27}$$

Note that both A and B_u are $N \times N$ sparse matrices. The problem can be efficiently solved by a variety of numerical packages. For instance, a standard function "eigs" in Matlab can be used to solve the above generalized eigenvalue problem.

We would like to point out that it is challenging to quantitatively characterize foldings, thus it is hard to theoretically predict the step number of the above folding removal iteration.

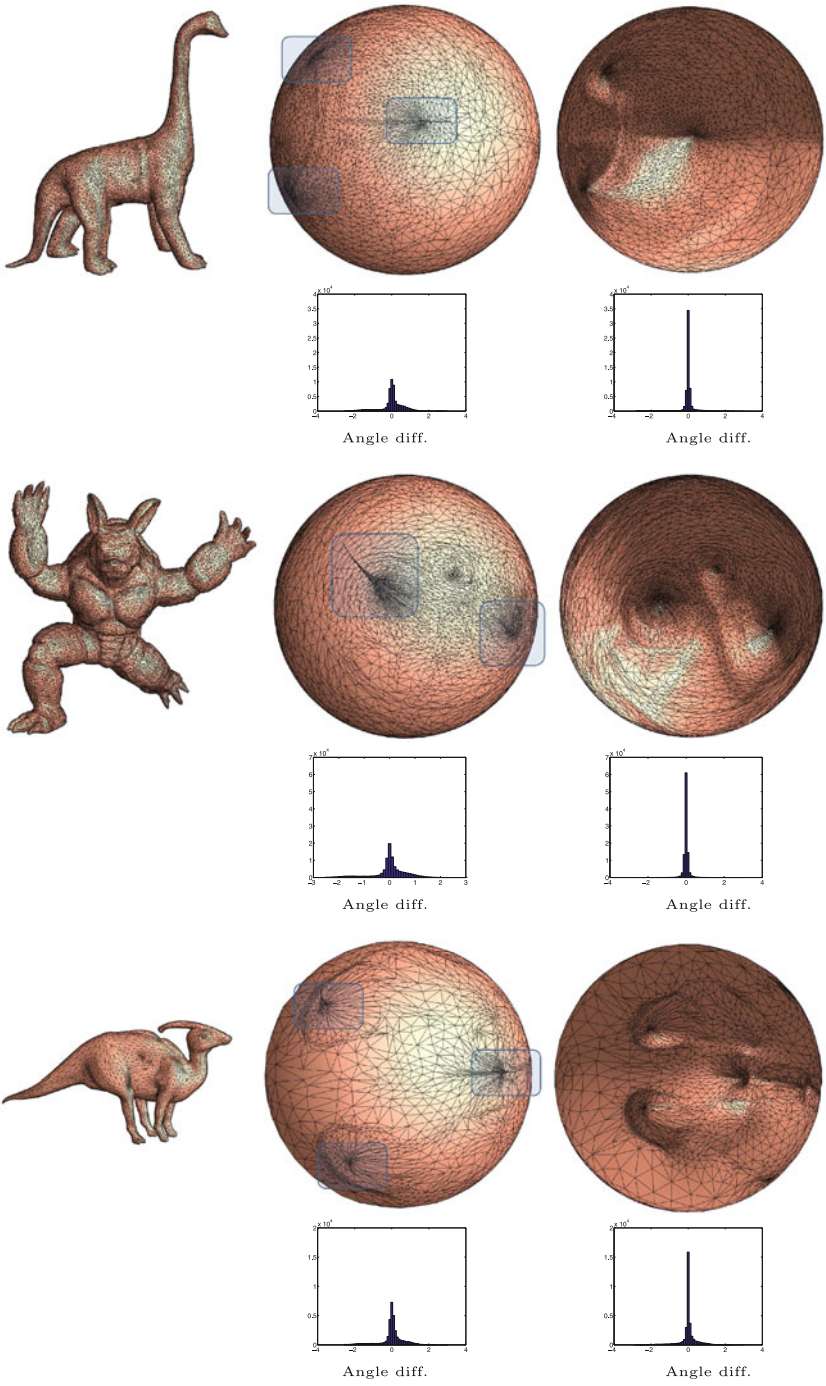

Fig. 10 Comparison of the results on Dino, Armadillo, Bird and Dilo surfaces obtained by Algorithm 2 and the algorithm in [1]. The *first column* the input surfaces; the *second column* the results of the algorithm in [1] with the histograms of angle differences; the *third column* the results of the proposed Algorithm 2 with the histograms of angle differences

Table 3 Comparison between the proposed Algorithm 2 and the algorithm in [1]

Surface	# of vertices	ϵ	The proposed Algorithm 2 (foldings removed)				Algorithm in [1] (foldings remained)	
			Iter 0 step	Iter 1 step	Iter 2 step	Time (s)	# of iterations	Time (s)
Dino	5,524	1e−10	4,480	1,518	3,652	91.11	2,854	835.53
Dilo	9,731	1e−10	3,610	5,000	4,184	73.94	3,106	513.56
Bird 1	950	1e−10	1,054	4,444	1,038	2.64	716	19.29
Bird 2	926	1e−10	592	1,674	5,000	14.44	599	14.09
Armadillo	16,519	1e−10	4,164	294	496	70.68	3,355	1,648.30

Generally speaking, the step number of the folding removal method is dependent on the complexity of the input surface. However, according to our experiments in Sect. 5 to surfaces with long and sharp features as complex as the shapes of dinosaur, armadillo and bird, only two steps of folding removal iteration will provide us folding free conformal maps.

5 Numerical Results

In this section, we demonstrate the efficiency and robustness of the proposed algorithms for computing global conformal maps of genus-0 surfaces to the unit sphere. On efficiency, we compare the proposed Algorithm 1 with the existing algorithm in [1]. On robustness against foldings, we extend the comparison to include the proposed Algorithm 2 on surfaces with long and sharp features. To further demonstrate the advantage of our global method, we also conduct comparison with a stereographic method introduced in [12]. All experiments were performed on a PC with a 2.66GHz CPU. It is worth noting that the algorithm [1] was written in C++, and our Algorithms 1 was implemented in MATLAB (Release 7.9.0) and the computation of weighted Laplace–Beltrami eigen-system was coded in C++. Since C++ is generally more efficient than MATLAB, the programming language difference does not introduce any biases toward our algorithms.

In our first experiment, we compare the speeds of Algorithm 1 and the algorithm [1] on several different surfaces listed in Fig. 8 with the corresponding Guass maps as initial maps. The number of iterations and computation times of both algorithms are given in Table 2. It is clear that Algorithm 1 is much more efficient on the given surfaces.

Our second experiment demonstrates how foldings are removed by Algorithm 2. The first row of Fig. 9 depicts the results of Algorithm 1 with obvious foldings and inaccurate conformal factors, which are highlighted by rounded boxes. Foldings are also observed in the results of the algorithm [1] given in the second column of Fig. 10. The second and third rows of Fig. 9 show the results after one and two folding removal iteration in Algorithm 2, respectively. It is clear that foldings are completely removed only using two iterations of folding removal iteration, which gives an accurate harmonic map.

To further demonstrate the robustness of Algorithm 2, we compute the harmonic maps obtained by Algorithm 2 and the algorithm [1] for a Dino surface, a Dilo surface, two bird surfaces and an armadillo surface. We also illustrate the corresponding histogram of angle differences of the resulting map $\mathbf{F}$. As we can expect, a good conformal map should have small angle differences. From the third column of Fig. 10, we can clearly see that the resulting maps

have good angle preserving properties. With only two steps of folding removal iteration, the results of Algorithm 2 are free of foldings. As comparisons, the results of the algorithm [1] are given on the second column of Fig. 10, and they contain foldings, thus the resulting angle differences are not satisfactory. In Table 3, we list the iteration numbers of each correction step and the total computing times of Algorithm 2, as well as those of the Algorithm [1]. In summary, the proposed Algorithm 2 can efficiently generate the folding-free harmonic maps.

6 Conclusions

This paper introduces an efficient algorithm for minimizing the harmonic energy problem, which quickly computes global conformal maps for genus-0 surfaces. To avoid foldings introduced by initial solutions which arise on surfaces with long and sharp features, a folding removal iteration based on the weighted Laplace–Beltrami eigen-projection is proposed. Numerical comparisons to the existing method [1] on several different surfaces demonstrate the efficiency and accuracy of the proposed algorithms. The generality of the proposed method can be further applied to solving variational problem of harmonic energy with additional fidelity terms or global p-harmonic minimization problems, which will be our future work.

Acknowledgments Rongjie Lai's work is supported by Zumberge Individual Award from USC's James H. Zumberge Faculty Research and Innovation Fund. Zaiwen Wen's work is supported in part by NSFC Grant 11101274 and Research Fund (20110073120069) for the Doctoral Program of Higher Education of China. Wotao Yin's work is supported in part by NSF Grant DMS-0748839 and ONR Grant N00014-08-1-1101. Xianfeng Gu's work is is supported in part of NSF Nets 1016286, NSF IIS 0916286, NSF CCF 1081424 and ONR N000140910228. Lok Ming Lui's work is supported in part of the CUHK Direct Grant (Project ID: 2060413) and HKRGC GRF (Project ID: 2130271).

References

1. Gu, X., Yau, S.: Computing conformal structures of surfaces. Commun. Inf. Syst. **2**(2), 121–146 (2002)
2. Gu, X., Wang, Y., Chan, T.F., Thompson, P., Yau, S.T.: Genus zero surface conformal mapping and its application to brain surface mapping. IEEE Trans. Med. Imaging **23**, 949–958 (2004)
3. Lui, L.M., Wang, Y., Thompson, P.M., Chan, T.F.: Landmark constrained genus zero surface conformal mapping and its application to brain mapping research. Appl. Numer. Math. **57**, 847–858 (2007)
4. Lui, L.M., Gu, X., Chan, T.F., Yau, S.-T.: Variational method on riemann surfaces using conformal parameterization and its applications to image processing. Methods Appl. Anal. **15**(4), 513–538 (2008)
5. Levy, B., Petitjean, S., Ray, N., Maillot, J.: Least squares conformal maps for automatic texture atlas generation. Proceeding of ACM SIGGRAPH (2002)
6. Eck, M., DeRose, T., Duchamp, T., Hoppe, H., Lounsbery, M., Stuetzle, W.: Multiresolution analysis of arbitrary meshes. Proceeding of ACM SIGGRAPH (1995)
7. Schoen, R., Yau, S.-T.: Lectures on Differential Geometry, vol. 2. International Press, Cambridge (1994)
8. Alliez, P., Meyer, M., Desbrun, M.: Interactive geometry remeshing. Proceeding of ACM SIGGRAPH (2002)
9. Kanai, T., Suzuki, H., Kimura, F.: Three-dimensional geometric metamorphosis based on harmonic maps. Vis. Comput. **14**(4), 166–176 (1998)
10. Hurdal, M.K., Stephenson, K., Bowers, P.L., Sumners, D.W.L., Rottenberg, D.A.: Coordinate systems for conformal cerebellar flat maps. NeuroImage **11**, S467 (2000)
11. Haker, S., Angenent, S., Tannenbaum, A., Kikinis, R., Sapiro, G., Halle, M.: Conformal surface parameterization for texture mapping. IEEE Trans. Vis. Comput. Graph. **6**(2), 181–189 (2000)
12. Springborn, B., Schröder, P., Pinkall, U.: Conformal equivalence of triangle meshes. ACM Transactions on Graphics (TOG)—Proceedings of ACM SIGGRAPH 2008, vol. 27(3) (2008)
13. Gu X., Yau S.T.: Global conformal surface parameterization. Symposium on Geometry Processing, pp. 127–137 (2003)

14. Jin M., Wang Y., Yau S.T., Gu X.: Optimal global conformal surface parameterization. IEEE Visualization, Austin, TX, pp. 267–274 (2004)
15. Jin, M., Kim, J., Luo, F., Gu, X.: Discrete surface ricci flow. IEEE Trans. Vis. Comput. Graph. **14**(5), 1030–1043 (2008)
16. Yang, Y., Kim, J., Luo, F., Hu, S., Gu, X.: Optimal surface parameterization using inverse curvature map. IEEE Trans. Vis. Comput. Graph. **14**(5), 1054–1066 (2008)
17. Schoen, R., Yau, S.-T.: Lectures on Harmonic Maps. International Press, Cambridge (1997)
18. Wen Z., Yin W.: A feasible method for optimization with orthogonality constraints. Math. Program. (2013, in press)
19. Lui, L.M., Thiruvenkadam, S., Wang, Y., Thompson, P.M., Chan, T.F.: Optimized conformal surface registration with shape-based landmark matching. SIAM J. Imaging Sci. **3**(1), 52–78 (2010)
20. Jost, J.: Riemannian Geometry and Geometric Analysis, 3rd edn. Springer, Berlin (2001)
21. Absil, P.-A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton, NJ (2008)
22. Helmke, U., Moore, J.B.: Optimization and Dynamical Systems. Communications and Control Engineering Series. Springer-Verlag London Ltd., London (1994). With a foreword by R. Brockett
23. Udrişte, Constantin: Convex Functions and Optimization Methods on Riemannian Manifolds, vol. 297 of Mathematics and its Applications. Kluwer Academic Publishers Group, Dordrecht (1994)
24. Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. IMA J. Numer. Anal. **8**(1), 141–148 (1988)
25. Zhang, H., Hager, W.W.: A nonmonotone line search technique and its application to unconstrained optimization. SIAM J. Optim. **14**(4), 1043–1056 (2004)
26. Meyer, M., Desbrun, M., Schröder, P., Barr, A.H.: Discrete differential-geometry operators for triangulated 2-manifolds. In: Hege, H.C., Polthier, K. (eds.) Visualization and Mathematics III, pp. 35–57. Springer, Berlin (2003)
27. Desbrun, M., Meyer, M., Schröder, P., Barr, A.H.: Discrete differential geometry operators in nd. In: Proceedings of VisMath'02, Berlin, Germany (2002)
28. Xu, G.: Convergent discrete Laplace–Beltrami operator over triangular surfaces. Proceedings of Geometric Modelling and Processing, pp. 195–204 (2004)
29. Chavel, I.: Eigenvalues in Riemannian Geometry. Academic Press, Inc., London (1984)
30. Ben-Chen, M., Gotsman, C.: Characterizing shape using conformal factors. Proceedings of Eurographics Workshop on Shape Retrieval, Crete, April (2008)
31. Reuter, M., Wolter, F.E., Peinecke, N.: Laplace–Beltrami spectra as Shape-DNA of surfaces and solids. Comput. Aided Des. **38**, 342–366 (2006)
32. Shi, Y., Lai, R., Krishna, S., Sicotte, N., Dinov, I., Toga, A.W.: Anisotropic Laplace–Beltrami eigenmaps: bridging reeb graphs and skeletons. In: Proceedings of MMBIA (2008)
33. Lai, R., Shi, Y., Scheibel, K., Fears, S., Woods, R., Toga, A.W., Chan, T.F.: Metric-induced optimal embedding for intrinsic 3D shape analysis. CVPR (2010)
34. Shi, Y., Lai, R., Gill, R., Pelletier, D., Mohr, D., Sicotte, N., Toga, A.W.: Conformal metric optimization on surface (CMOS) for deformation and mapping in Laplace–Beltrami embedding space. MICCAI (2011)