SIAM J. Imaging Sciences
Vol. 6, No. 1, pp. 78–101

Conformal Mapping with as Uniform as Possible Conformal Factor*

Yonathan Aflalo†, Ron Kimmel‡, and Michael Zibulevsky‡

Abstract. According to the uniformization theorem, any surface can be conformally mapped into a domain of a constant Gaussian curvature. The *conformal factor* indicates the local scaling introduced by such a mapping. This process could be used to compute geometric quantities in a simplified flat domain with zero Gaussian curvature. For example, the computation of geodesic distances on a curved surface can be mapped into solving an eikonal equation in a plane weighted by the conformal factor. Solving an eikonal equation on the weighted plane can then be done by regular sampling of the domain using, for example, the celebrated *fast marching method* (FMM). The connection between the conformal factor on the plane and the surface geometry can be justified analytically. Still, in order to construct consistent numerical solvers that exploit this relation, one needs to prove that the conformal factor is bounded. We provide theoretical bounds of the conformal factor and introduce optimization formulations that control its behavior. It is demonstrated that without such restrictions the numerical results are unboundedly inaccurate. Putting all ingredients in the right order, we introduce a method for computing geodesic distances on a two-dimensional manifold by using the FMM on a weighted flat domain. It is also shown how a metric on a curved domain can be reconstructed by reformulating the nonflat metric restoration problem into a weighted flat domain, again, with bounded weights for consistent results.

Key words. conformal mapping, geodesics computation, optimal conformal mapping, cotangent weights

AMS subject classification. 30C30

DOI. 10.1137/110845860

1. Introduction.

Consistent and efficient distance computation on various domains is a key component in many important applications. Several papers tackle the problem of geodesic distance computation on triangulated surfaces. The celebrated *fast marching method* (FMM) [18, 22] enabled the solution in isotropic inhomogeneous domains that are regularly sampled. It was later generalized [11] through a geometric interpretation of the numerical update step, which enabled consistent and efficient computation of distances in anisotropic domains. So far, the FMM has been implemented on manifolds given as either a triangulated mesh or a parametrized surface [23, 20], or implicitly defined in a narrow band numerically sampled with a regular grid [15]. Traditionally, the FMM is executed on the manifold itself where some parametrization is provided. In these cases, usually there is some processing involved in order to overcome the irregularity of the numerical sampling. This is, for example, the case for the unfolding initialization step in [11]. Here, in order to avoid this procedure, we use a conformal

*Received by the editors August 25, 2011; accepted for publication (in revised form) August 30, 2012; published electronically January 29, 2013. This research was supported by the European Community's FP7-ERC program, grant agreement 267414.
http://www.siam.org/journals/siims/6-1/84586.html
†Faculty of Electrical Engineering, Technion University, Haifa 32000, Israel (yaflalo@cs.technion.ac.il).
‡Faculty of Computer Science, Technion University, Haifa 32000, Israel (ron@cs.technion.ac.il, mzib@cs.technion.ac.il).

mapping of a given surface and compute distances in a simplified domain. The conformal mapping has been used for several applications, such as shape correspondences [14], surface comparisons [13, 8], and surface matching [6]. It can be used to map one surface into another such that the pull-back metric of the mapping is proportional to the original metric scaled by a scalar called the conformal factor. In particular, the mapping space can be chosen to be flat. Quasi-conformal mappings have also been used for surface analysis [24]. Though useful for some applications, departing from conformal mapping distorts the local isotropy of the metric and thus, as an example, would prohibit us from using simple eikonal solvers on a uniformly sampled flat domain in order to compute geodesics. In this paper, we conformally map the original curved surface into a flat plane in which we run the FMM using the conformal factor as a local weight.

The paper is organized as follows. In section 2, we introduce the Laplace–Beltrami operator that allows us to define the conformal mapping in section 3. Section 4 shows the results of the weighted FMM on the conformal mapping. Section 5 presents an optimization scheme that makes the conformal mapping as uniform as possible, enabling us to use the FMM algorithm. In section 6 we extend this technique to zero-genus shapes by conformally mapping to the sphere. Then, in section 7 we justify theoretically the legitimacy and consistency of our method and describe an application of the proposed technique to the anisotropic metric reconstruction problem.

2. Introduction to the Laplace–Beltrami operator. The Laplace–Beltrami operator is an extension of the Laplacian to nonflat multidimensional manifolds. Its interesting properties have been studied and used extensively in computer graphics, differential geometry, and shape analysis. Given a manifold $\mathcal{S}$ defined by a parametrization form $f(u) : \mathbb{R}^n \to \mathbb{R}^m$, we can introduce a metric derived from the so-called first fundamental form of f. When moving a distance $\|du\|$ in $\mathbb{R}^n$, we can easily show that we move a distance equal to $du^T J^T J du = du^T G du$ in $\mathcal{S}$, where J represents the Jacobian of f, and G is the induced metric. We note that the matrix G is positive definite and symmetric by definition. The scalar product induced by the metric G is given by

$$\langle u, v \rangle_G = u^T G v = v^T G u.$$

This scalar product implies a new definition of the gradient induced by G, $\nabla_G f$ defined as

$$\begin{aligned} f(u+du) &= f(u) + \langle \nabla_G f, du \rangle_G + o(\|du\|) = f(u) + \nabla_G f^T G du + o(\|du\|) \\ &= f(u) + \nabla f^T du + o(\|du\|). \end{aligned}$$

We readily have

$$\nabla_G f = G^{-1} \nabla f.$$

Let us now introduce the Laplace–Beltrami operator. While working in $\mathbb{R}^n$ with the usual Laplace operator defined by $\Delta f = \sum_{i=1}^n \partial_i \partial_i f$, an interesting property of the Laplacian is

$$\int_{u \in \Omega} \Delta f g da = - \int_{u \in \Omega} \langle \nabla f, \nabla g \rangle da \quad \forall g | g = 0 \text{ on } \partial\Omega,$$

where da is the infinitesimal area element. Hence, a natural extension of the Laplacian for a given scalar product and a given definition of the gradient would consist of finding $\Delta_G f$ such

that

$$\int_{u\in\Omega} \Delta_G f h da = -\int_{u\in\Omega} \langle \nabla_G f, \nabla_G h\rangle_G \sqrt{g} du_1 \dots du_n \quad \forall h \text{ such that } h_{|\partial\Omega} = 0,$$

where $g = \det(G)$ and $da = \sqrt{g} du_1 \dots du_n$ is the local infinitesimal area element according to the metric G. We have

$$\begin{aligned}\int_{u\in\Omega} \langle \nabla_G f, \nabla_G h\rangle_G \sqrt{g} du_1 \dots du_n &\equiv \int_{u\in\Omega} \nabla f^T G^{-1} \nabla h \sqrt{g} du_1 \dots du_n \\ &= \int_{u\in\Omega} \left(\sqrt{g} G^{-1} \nabla f\right)^T \nabla h du_1 \dots du_n\end{aligned}$$

since the matrix G is symmetric. Integration by parts leads to

$$\int_{u\in\Omega} \left(\sqrt{g} G^{-1} \nabla f\right)^T \nabla h du_1 \dots du_n = -\int_{u\in\Omega} \left(\sum_{i=1}^{n} \partial_i \left(\sqrt{g} G^{-1} \nabla f\right)_i h\right) du_1 \dots du_n.$$

We obtain

$$\begin{aligned}&\int_{u\in\Omega} \left(\sum_{i=1}^{n} \partial_i \left((\sqrt{g} G^{-1}) \nabla f\right)_i h\right) du_1 \dots du_n \\ &= \int_{u\in\Omega} \Delta_G f h \sqrt{g} du_1 \dots du_n \quad \forall h \text{ such that } h_{|\partial\Omega} = 0.\end{aligned}$$

It follows that

$$\Delta_G f = \frac{1}{\sqrt{g}} \sum_{i=1}^{n} \partial_i \left((\sqrt{g} G^{-1}) \nabla f\right)_i.$$

Using Einstein summation convention, given a metric G on a manifold, we define the Laplace–Beltrami operator as

$$\Delta_G f = \frac{1}{\sqrt{g}} \partial_i \left(\sqrt{g} g^{ij} \partial_j f\right),$$

where g is the determinant of G, $g^{ij} = \left(G^{-1}\right)_{i,j}$, and ∂_i is the derivative with respect to the ith coordinate.

2.1. Introduction to conformal mapping. Let us consider a two-dimensional parametrized manifold $\mathcal{X} \in \mathbb{R}^3$. It can be defined by the functions $x, y, z : \mathbb{R}^2 \to \mathbb{R}$, such that the parametrization $(\alpha, \beta) \in \mathbb{R}^2$ defines a coordinate in $\mathcal{X}$ given by $\mathcal{X} = (x(\alpha, \beta), y(\alpha, \beta), z(\alpha, \beta))$. Such a parametrization induces a metric G, a scalar product $\langle u, v\rangle_G = u^T G v$, a gradient $\nabla_G \cdot = G^{-1} \nabla \cdot$, where $\nabla \cdot$ is the usual gradient with respect to α and β, and a Laplace–Beltrami operator $\Delta_G \cdot = 1/\sqrt{g} \nabla^T \left(\sqrt{g} G^{-1} \nabla \cdot\right)$, where $g = \det(G)$. We would like to map the surface $\mathcal{X}$ into $D \in \mathbb{R}^2$, preserving the angles of intersections of corresponding curves. That is, given any two curves in $\mathcal{X}$, their images in D have to intersect at the same angle as in $\mathcal{X}$. A conformal mapping is a mapping function that has this property at each and every point

and can be defined by two functions $(u(\alpha,\beta), v(\alpha,\beta))$ that map our manifold in D and obey the condition

$$\nabla u = \frac{GR}{\sqrt{g}}\nabla v,$$

where $R = \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$. The matrix R satisfies the following equalities:

$$\begin{aligned} GRG &= \det(G)R \qquad \forall G|G^T = G, \\ GR^TG &= \det(G)R^T \qquad \forall G|G^T = G, \\ RGR^T &= \det(G)G^{-1} \qquad \forall G|G^T = G. \end{aligned}$$

This restriction over (u, v) implies the following properties:

Property 1.

$$\begin{aligned} \Delta_G u &= \frac{1}{\sqrt{g}}\nabla^T \cdot \left(\sqrt{g}G^{-1}\nabla u\right) = \frac{1}{\sqrt{g}}\nabla^T \cdot \left(\sqrt{g}G^{-1}\frac{GR}{\sqrt{g}}\nabla v\right) \\ &= \frac{1}{\sqrt{g}}\nabla^T \cdot (R\nabla v) = \frac{1}{\sqrt{g}}\underbrace{(\nabla^T \cdot R\nabla)}_{=0} v = 0. \end{aligned}$$

Property 2.

$$\Delta_G v = \frac{1}{\sqrt{g}}\nabla^T \cdot \left(g\underbrace{G^{-1}R^TG^{-1}}_{=\frac{1}{g}R^T}\nabla u\right) = \frac{1}{\sqrt{g}}\underbrace{(\nabla^T \cdot R^T\nabla)}_{=0} u = 0.$$

Property 3.

$$\langle \nabla_G u, \nabla_G v\rangle_G = \nabla v^T G^{-1}\nabla u = \frac{1}{\sqrt{g}}\nabla v^T R\nabla v = 0$$

since R is a $\frac{\pi}{2}$-rotation angle.

Property 4.

$$\begin{aligned} \langle \nabla_G u, \nabla_G u\rangle_G &= \nabla u^T G^{-1}\nabla u = \frac{1}{g}\nabla v^T \underbrace{R^TGG^{-1}GR}_{=R^TGR=gG^{-1}}\nabla v = \nabla v^T G^{-1}\nabla v \\ &= \langle \nabla_G v, \nabla_G v\rangle_G. \end{aligned}$$

This is equivalent to the Cauchy–Riemann condition if we take the metric $G = I$.

Denoting by J the Jacobian of the mapping $(\alpha,\beta) \to (u, v)$, the previous conditions can be written as

$$\begin{aligned} \begin{pmatrix} \|\nabla_G u\|_G^2 & \langle \nabla_G u, \nabla_G v\rangle_G \\ \langle \nabla_G u, \nabla_G v\rangle_G & \|\nabla_G v\|_G^2 \end{pmatrix} &= \|\nabla_G u\|_G^2 I \\ &\Leftrightarrow (\nabla_G u, \nabla_G v)^T G(\nabla_G u, \nabla_G v) = \|\nabla_G u\|_G^2 I \\ &\Leftrightarrow (\nabla u, \nabla v)^T G^{-1}(\nabla u, \nabla v) = \|\nabla_G u\|_G^2 I \\ &\Leftrightarrow JG^{-1}J^T = \|\nabla_G u\|_G^2 I \\ &\Leftrightarrow G^{-1} = \|\nabla_G u\|_G^2 J^{-1}J^{-T} \\ &\Leftrightarrow J^TJ = G\|\nabla_G u\|_G^2. \end{aligned}$$

Hence, any mapping is conformal with respect to a metric G if and only if there exists a scalar function μ, referred to as the *conformal factor*, such that its Jacobian J satisfies $J^TJ = \mu^2 G$. We also note that

$$\left\| \begin{pmatrix} du \\ dv \end{pmatrix} \right\|^2 = \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix}^T J^T J \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix} = \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix}^T \mu^2 G \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix} = \mu^2 \left\| \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix} \right\|_G^2.$$

It follows that

$$\left\| \begin{pmatrix} d\alpha \\ d\beta \end{pmatrix} \right\|_G = \frac{1}{\mu} \left\| \begin{pmatrix} du \\ dv \end{pmatrix} \right\|.$$

Such a mapping would allow us to compute distances on any two-dimensional metric space with a generalized metric G using the computation of distance in an inhomogeneous isotropic flat manifold.

3. Construction of a discrete harmonic map. We start with a theorem that would be useful for our conformal map construction.

Theorem 3.1. *Given a metric G defined on a regular domain D, and a function f defined on ∂D, the solution f of the problem*

$$\underset{\substack{f\in C^2(D)\\ f(x)=f_0(x)\ \forall x\in\partial D}}{\operatorname{argmin}} \left\{ \int_D \|\nabla_G f\|_G^2 \right\}$$

satisfies $\Delta_G f = 0$ and $f(x) = f_0(x)$ for all $x \in \partial D$.

The main idea when constructing a discrete conformal map according to Polthier [16] is to find a triangulation $\mathfrak{T} = \{T_1, \ldots, T_{N_T}\}$ (where T_i is a triangle and N_T is the number of triangles) of our map with N_V vertices, and search for a continuous function $u \in C(D)$ minimizing the Dirichlet energy defined in Theorem 3.1. For example, we could find u given by $u(\gamma) = u_0(\gamma) + \sum_{i=1}^{N_V} u_i \phi_i(\gamma)$, where u_i are some coefficients and ϕ_i are functions satisfying the following:

1. $\phi_i \in C^0(D)$, where $C^0(D)$ represents the space of continuous functions defined on D.
2. $\phi_i(V_j) = \delta_{ij}$ for all $i, j \in \{1, \ldots, N_V\}$ and

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

3. ϕ_i is linear in each triangle.

V_j here denotes the jth vertex of $\mathfrak{T}$.

Considering a triangle T with the vertices (V_p, V_q, V_r) and the respective angles $(\theta_p, \theta_q, \theta_r)$, we show that the following hold:

1. $\langle \nabla\phi_p, \nabla\phi_q \rangle = -\frac{\cot\theta_r}{2 \text{ area } T}$.
2. $\|\nabla\phi_q\|^2 = \frac{\cot\theta_p + \cot\theta_q}{2 \text{ area } T}$.

The algorithm then minimizes the Dirichlet energy of the function u with respect to u_i:

$$\begin{aligned} E(u) &= \frac{1}{2}\int_{\mathfrak{T}} \left\| \nabla u_0(x) + \sum_{i=1}^{N_v} u_i \nabla \phi_i(x) \right\|^2 dx \\ &= \frac{1}{2}\sum_{T_i \in \mathfrak{T}} \int_{T_i} \|\nabla t\|^2 dx \\ &= \frac{1}{4}\sum_{\text{edges } (i,j)} (\cot\theta_{ij} + \cot\psi_{ij})(u_i - u_j)^2, \end{aligned}$$

where $u_j = u(V_j)$ and θ_{ij} and ψ_{ij} represent the angles supporting the edge V_iV_j, where V_j is a neighbor of V_i, $u_i = u(V_i)$, and

$$\frac{\partial E(u_{|_{T_j}})}{\partial u_i} = \frac{1}{2}\sum_{\text{edges } (i,j) \text{ at } i} (\cot\theta_{ij} + \cot\psi_{ij})(u_i - u_j) = 0$$

at the optimality. We then have to solve the following system of equations to find a harmonic function u:

$$\sum_{\text{edges } (i,j) \text{ at } i} (\cot\theta_{ij} + \cot\psi_{ij})(u_i - u_j) = 0 \quad \forall i. \tag{3.1}$$

After u has been computed, we have to find another conjugate discrete harmonic function v such that $\nabla v = \frac{GR}{\sqrt{g}}(\nabla u)$. In other words, we have to compute the gradient of u and perform a rotation by $\frac{\pi}{2}$. To achieve this goal, Polthier [16] proposed defining a midedge grid. For each edge (V_i, V_j), define a vertex at the middle of the edge as $V_s^* = \frac{V_i+V_j}{2}$. This way, each triangle (V_1, V_2, V_3) is associated with a new triangle (V_1^*, V_2^*, V_3^*). If we define Ψ_r, the function associated to the vertex V_r^* in the midedge grid (or, equivalently, to the edge (V_i, V_j) in the regular grid), one can show that

$$\begin{pmatrix} v_3 - v_1 \\ v_3 - v_2 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} (u_2 - u_1)\cot\theta_{21} + (u_2 - u_3)\cot\theta_{23} \\ (u_2 - u_1)\cot\theta_{21} + (u_3 - u_1)\cot\theta_{31} \end{pmatrix},$$

where v_r, v_s are the values of v at the midedge vertices V_r^*, V_s^* located along the edges $(V_i, V_j), (V_j, V_k)$, respectively, and θ_{jk} is the oriented angle supporting the edge (j, k).

We end up with an algorithm, summarized, for example, in [14, 16], that computes the midedge conformal flattening.

Algorithm 1. Midedge discrete conformal map.

Require: $\mathfrak{T}$ triangulation of the space Ω

Choose a face to cut, $C = \{V_{i_c}, V_{j_c}, V_{k_c}\} \in \mathfrak{T}$, and solve

$$\sum_{j \in \mathcal{N}(i)} (u_i - u_j)\left(\cot\theta_{ij} + \cot\psi_{ij}\right) = 0 \quad \forall i \notin \{i_c, j_c, k_c\}.$$

Set arbitrary value for u on C and solve

$$\begin{pmatrix} v_j - v_k \\ v_j - v_l \end{pmatrix} = \frac{1}{2}\begin{pmatrix} (u_l - u_k)\cot\theta_{lk} + (u_l - u_j)\cot\theta_{lj} \\ (u_l - u_k)\cot\theta_{lk} + (u_j - u_k)\cot\theta_{jk} \end{pmatrix}.$$

For the midedge vertex $V_r^* = \frac{V_p + V_q}{2}$, set the value of the conformal map on the midedge grid

$$u_r^* = \frac{u_p + u_q}{2}, \quad v_r^* = v_r.$$

We also have the value of the conformal factor for each triangle $T_k = (V_p, V_q, V_r)$, given by

$$\begin{aligned} \mu(T_k) &= \|\nabla u(x_q)\| \\ &= \left(\frac{(u_r - u_q)^2\cot\theta_p + (u_p - u_q)^2\cot\theta_r + (u_r - u_p)^2\cot\theta_q}{2 \text{ area } T_q}\right)^{\frac{1}{2}}, \end{aligned} \tag{3.2}$$

which is nothing but the area ratio of the triangles before and after the mapping.

4. Fast marching on the conformal map. Conformally mapping a curved surface into the plane, while restricting the amount of the local scaling, allows us to translate the geodesic distance computation on curved (anisotropic) homogeneous domains to a similar problem in flat nonhomogeneous isotropic domains. This new setting of the problem allows us, for example, to use regular grids for numerically computing geodesic distances.

There are several methods for computing distances on surfaces. There is the so-called *exact* method, according to which the geodesics on the triangulated polyhedron approximating the surface are computed in an exact manner. This is, in fact, a second order accurate method for which the practical complexity is about $O(n^2)$, where n is the number of vertices of the triangulated surface [12, 3, 23]. At the other end there exist second order accurate methods that compute distances on weighted flat domains that are regularly sampled in about $O(n)$; see, e.g., Sethian [19]. This second order method, operating on the conformal plane, allows us to gain one order of magnitude in either complexity or accuracy compared to the exact methods. Surazhsky and Surazhsky [21] propose also a fast exact and approximate geodesic computation on meshes.

In the following experiments, we conformally map several functions into $\mathbb{R}^2$ and run the fast marching algorithm on the conformal map using the conformal factor as a local scaling of a uniform isotropic metric tensor. That is, we numerically solve the eikonal equation $||\nabla U(x,y)|| = \mu(x,y)$. When mapping a surface, we have to take care of the boundary

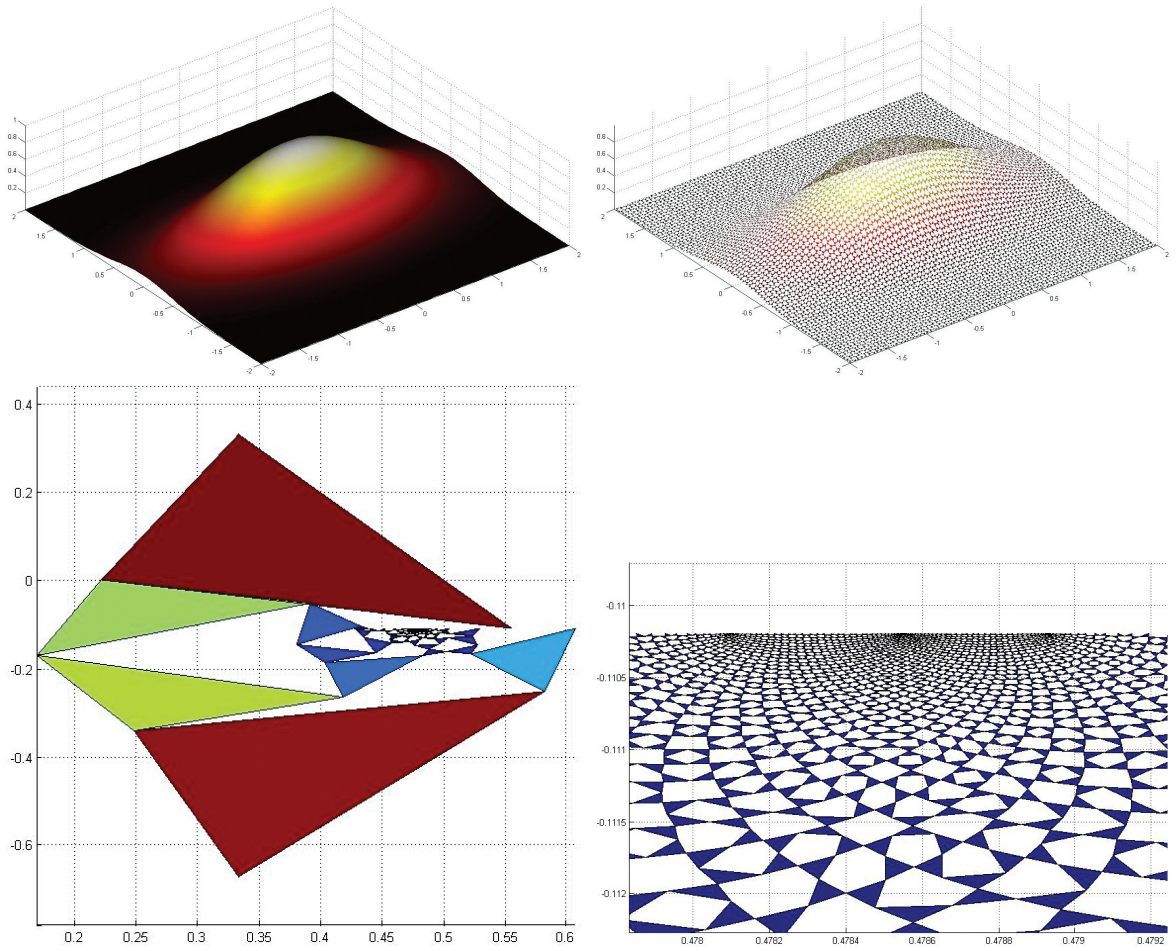


Figure 1. *Left to right, top to bottom: Original surface, midedge surface, conformal map, and zoom-in.*

conditions. The way we define the boundary of our target map is important. It can help us control the conformal factor and thereby the numerical accuracy of our scheme. Without controlling the boundary, all the points on the surface boundary could be mapped to a line. While uniforming the metric and solving one problem, we encounter a new one, that is, a nonuniform conformal factor. The conformal factor observes the curvature of the surface on one hand, but yields a challenging, highly nonuniformly sampled numerical domain on the other.

In our first example, Figure 1, we map the surface $z = f(x, y) = \exp(-0.2x^2 - 0.5y^2)$ without controlling the boundary.

If we zoom in the area with the smallest triangles, we observe that there are three points around which small triangles are concentrated. These points correspond to the corners of the original surface. When we compute the geodesic distances from the corner point $(-2, -2)$ to the rest of the surface points, the result presented in Figure 2 demonstrates numerical inaccuracies due to variability of the conformal factor.

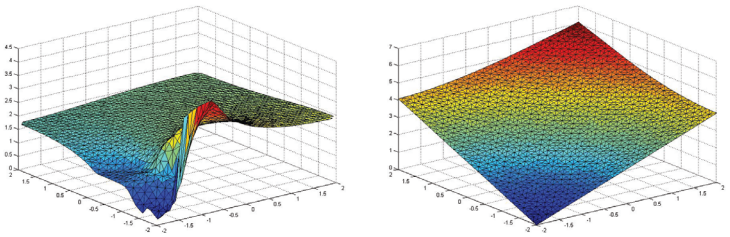


Figure 2. *Geodesic distance from the point* $(-2,-2)$ *computed with FMM on the conformal map (left) and with the FMM on the triangulated domain (right).*

Our next challenge would be to bound the ratio between the smallest conformal factor and the largest one. Actually, in the above example, the areas ratio is on the order of 10^{-13}, and the conformal factor ratio is 10^{-7}. Therefore, it is impractical to numerically approximate geodesic distances using the FMM on the uniform grid obtained by sampling an arbitrary conformal map. Next, we try to overcome this difficulty by manipulating the location of boundary points of the conformal map.

5. Controlling the conformal factor.

5.1. Original formulation. Jin et al. [9] optimized an integral over the conformal factor of a zero-genus surface conformally mapped to a sphere. They exploit the Moebius group of transformations that links between all conformal mappings in such a setting. Here, we map a given surface into a bounded domain in which we would like to maximize the minimal conformal factor. To do this, we start by studying the computational aspect of the problem. We could try to manipulate the boundary conditions. In Polthier's algorithm, the scheme involves finding u and v. We find u by solving the system of equations (3.1). More precisely, this system of equations is defined for each vertex i that does not belong to the boundary. Define L to be the matrix of cotangent weights such that the previous equations can be written as $Lu = 0$. Make B the set of indices of the points along the boundary. Let us define $\tilde{L}$ to be the matrix obtained by replacing in L the rows that correspond to boundary points with the rows of the identity matrix taken at these indices. We also introduce P, a matrix defined by

$$P_{i,j} = \begin{cases} 0 & \text{if } i \notin B, \\ 1 & \text{if } i \in B \text{ and } i \text{ is the } j\text{th vertex in } B, \end{cases}$$

and $\tilde{u}$, a vector representing the ordered values of u along the boundary. $\tilde{u}$ contains the u_i for $i \in B$.

We can show that the first coordinate u of the conformal mapping obeys the following relation:

$$Lu = P\tilde{u}.$$

We can also show from (3.2) that for each triangle T_i, there exists a sparse matrix denoted by K_i, such that

$$\mu(x_i) = \sqrt{u^T K_i u}.$$

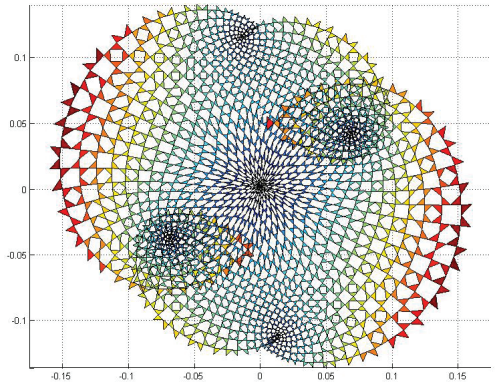


Figure 3. *Unconstrained optimal conformal map.*

We would like to control the ratio between the smallest conformal factor and the largest one. We do so by maximizing the following expression:

$$\begin{aligned}&\max_{u_j,u} \frac{\min_i \mu(x_i)^2}{\max_i \mu(x_i)^2}\\ &\text{subject to (s.t.)}\\ &u_j \in [0,1] \quad \forall j \in B.\end{aligned}$$

The above problem can be reformulated as

$$\begin{aligned}&\max_{\tilde{u}_j,u} \left[\frac{\min_i u^T K_i u}{\max_i u^T K_i u}\right]\\ &\text{s.t.}\\ &Lu = P\tilde{u},\\ &\tilde{u}_j \in [0,1] \quad \forall j.\end{aligned}$$

Since $\tilde{u}$ represents the first coordinate of the boundary points, to avoid foldovers we have to make sure that the boundary coordinates are increasing and decreasing at most once. The coordinates of $\tilde{u}$ have to grow up to an index from which they decrease. Without loss of generality, we choose this index to be $\frac{\#B}{2}$. This constraint can be written as

$$A\tilde{u} \leqslant 0,\ A = \begin{pmatrix} 1 & -1 & 0 & \cdots & \cdots & \cdots \\ 0 & 1 & -1 & 0 & \cdots & \cdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ 0 & \cdots & -1 & 1 & 0 & \cdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ 0 & \cdots & \cdots & \cdots & -1 & 1 \end{pmatrix}.$$

Actually, without the previous constraint, we could get a conformal map with foldovers as shown in Figure 3.

The optimization problem can be rewritten as

$$\begin{aligned}
&\max_{\tilde{u}_j,u} \left[\frac{\min_i u^T K_i u}{\max_i u^T K_i u}\right] \\
&\text{s.t.} \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0.
\end{aligned}$$

Since the conformal factor that represents the local metric is upper bounded by the area of the target map, we can solve the following problem:

$$\begin{aligned}
&\max_{\tilde{u}_j,u} \min_i u^T K_i u \\
&\text{s.t.} \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0.
\end{aligned} \tag{5.1}$$

For this goal, since Problem (5.1) is not convex, we present three variations of the above formulation, two of which involve convex programming.

5.2. Linear programming relaxation. Let us define the dual of (5.1):

$$\begin{aligned}
&\min_i \max_{\tilde{u}_j,u} u^T K_i u \\
&\text{s.t.} \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0.
\end{aligned} \tag{5.2}$$

We can divide the problem into decoupled subproblems. The matrices K_i are symmetric and thus admit a spectral decomposition $K_i = P_i^T D_i P_i$, where D_i are diagonal and $P_i^T P_i = \mathbf{1}$. Denoting by $v = P_i u$, we have $v^T D_i v = \sum_j D_j v_j^2$, which leads to the following problem:

$$\begin{aligned}
&\max_{\tilde{u}_j,v,u} \sum_j D_j v_j^2 \\
&\text{s.t.} \\
&v = P_i u, \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0.
\end{aligned}$$

Replacing the expression $\sum_j D_j v_j^2$ with $\sum_j D_j v_j$ provides us with a linear programming problem that we solve for all triangles T_i.

5.3. Semidefinite programming relaxation. Next, we try to solve (5.2) using a semidefinite relaxation. We first notice that $u^T K_i u = \operatorname{trace}(K_i u u^T)$. Problem (5.2) can be equivalently written as

$$\begin{aligned}
&\min_i \max_{\tilde{u}_j,\tau,U,u} \tau \\
&\text{s.t.} \\
&\operatorname{trace}(K_i U) \leqslant \tau, \\
&U = uu^T, \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0.
\end{aligned}$$

Since we want to maximize τ, we can relax the problem and replace the constraint $U = uu^T$ with $U \leqslant uu^T$, which, according to the Schur complement, is rigorously equivalent to

$$\begin{pmatrix} U & u \\ u^T & 1 \end{pmatrix} \leqslant 1.$$

Our problem can thereby be approximated by

$$\begin{aligned}
&\min_i \max_{\tilde{u}_j,\tau,U,u} \tau \\
&\text{s.t.} \\
&\operatorname{trace}(K_i U) \leqslant \tau, \\
&\begin{pmatrix} U & u \\ u^T & 1 \end{pmatrix} \leqslant 1, \\
&Lu = P\tilde{u}, \\
&\tilde{u}_j \in [0,1] \quad \forall j \in B, \\
&A\tilde{u} \leqslant 0,
\end{aligned}$$

that is, a discrete minimum over a set of semidefinite programming problems.

5.4. Smoothing reformulation. The minimization over a discrete set can be approximated using a sequence of functions whose limit converges to the minimum of a given set of reals. Let us consider a vector $x \in \mathbb{R}^n$. We can define the soft maximum as a parametric function $f_p : \mathbb{R}^n \to \mathbb{R}$ defined by

$$f_p(x) = \left(\sum_{i=1}^{n} x_i^{-p} \right)^{-\frac{1}{p}},$$

and the soft minimum as

$$h_p(x) = \left(\sum_{i=1}^{n} x_i^{p} \right)^{\frac{1}{p}}.$$

It is easy to see [2] that $\lim_{p\to\infty} f_p(x) = \min_i x_i$ and that $\lim_{p\to\infty} h_p(x) = \max_i x_i$. We use this relaxation to solve the following optimization program:

$$\begin{aligned}
&\max_{\tilde{u}_j,u,x} \frac{f_p(x)}{h_p(x)}\\
&\text{s.t.}\\
&x_i = u^T K_i u \quad \forall i,\\
&Lu = P\tilde{u},\\
&\tilde{u}_j \in [0,1] \quad \forall j \in B,\\
&A\tilde{u} \leqslant 0,
\end{aligned}$$

which can be rewritten as

$$\begin{aligned}
&\max_{\tilde{u}_j,u,x} \frac{f_p([u^tK_1u, u^tK_2u, \ldots, u^tK_nu])}{h_p([u^tK_1u, u^tK_2u, \ldots, u^tK_nu])}\\
&\text{s.t.}\\
&Lu = P\tilde{u},\\
&\tilde{u}_j \in [0,1] \quad \forall j \in B,\\
&A\tilde{u} \leqslant 0
\end{aligned} \tag{5.3}$$

for a reasonable value of p. The main advantage of this technique compared to the two previous formulations is the fact that we do not need to couple the solutions of separate optimization problems. The disadvantage is the fact that this problem is not convex. Thus, we have to consider a good initialization based on which we can use a penalization method to reach the minimum.

5.5. A sparsity dilemma. The above problems involve only sparse matrices, but the variable vectors lie in high-dimensional space. We can reduce the dimensionality of the problem by eliminating the vector u, but this would imply working with nonsparse matrices. For this goal, we could use the fact that $Lu = P\tilde{u}$ and write $u = (L\backslash P)\tilde{u} = \tilde{P}\tilde{u}$. Then, using the matrix $\tilde{K}_i = \tilde{P}^T K_i \tilde{P}$, we have $u^T K_i u = \tilde{u}^T \tilde{K}_i \tilde{u}$. We are led to a lower-dimensional problem as we have lost the advantage of sparsity matrices.

5.6. Results. We solve problem (5.3) with the penalty/barrier multiplier (PBM) method [2] using the optimization toolbox [25]. The areas ratio in our example can be increased from 0.001 to 0.34 and the conformal factor ratio changed from 0.003 to 0.59. We can then obtain accurate numerical results while computing the geodesic distances as seen in Figure 4 and can compare the error between consistent geodesic distances (computed with the Tosca toolbox [4]) and the geodesic distances computed with FMM on a flat, regularly sampled domain. We notice, in this case, that the error is of the same order as that of the FMM computed on the triangulation [11].

We repeat the experiment for a scanned face and compute the optimal mapping of the surface.

So far, we have demonstrated the difficulties of working with conformal mapping and have shown that manipulating the boundary conditions can lead to a consistent scheme. Gu's L_2 setting [9] is more forgiving to vanishing features as this is an integral measure, while our

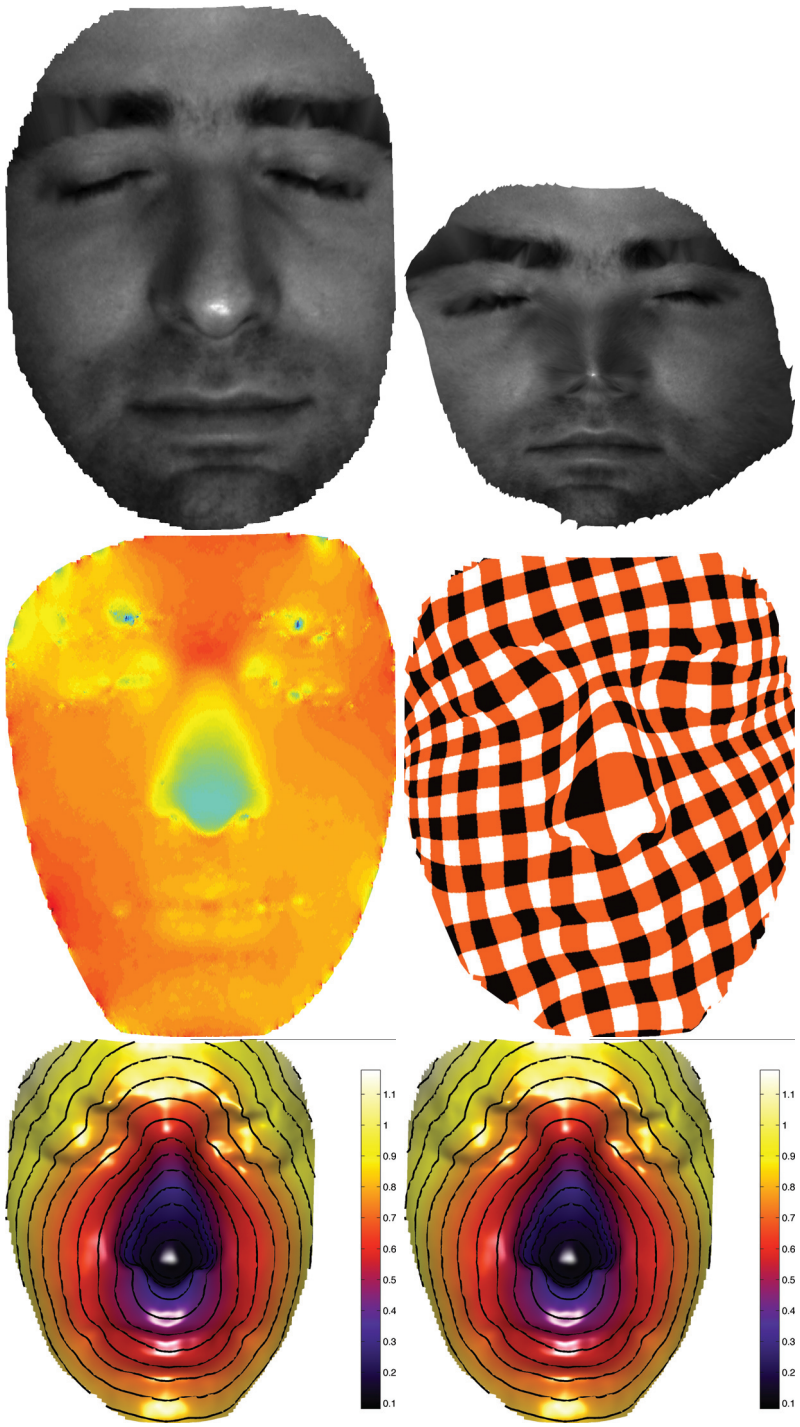


Figure 4. *Left to right, top to bottom: Original surface, conformal map optimized for* $\max \frac{\mu_{min}}{\mu_{max}}$*, conformal factor, texture mapping on the conformal map mapped back onto the original surface, geodesic distance with FMM on the surface, and geodesic distance with FMM on the conformal map.*

L_∞ norm better captures deviation from exceptions that usually happen at regions of interest (high curvature) that are significant in three-dimensional object analysis. Next, we provide more motivation for maximizing the minimal conformal factor in a surface topologically homeomorphic to a sphere, that is, a domain without boundary.

6. Conformal mapping onto a sphere. Consider a sphere with the following parametrization:

$$\begin{aligned} x &= R\cos\theta\cos\phi, \\ y &= R\sin\theta\cos\phi, \\ z &= R\sin\phi. \end{aligned} \tag{6.1}$$

The corresponding induced metric can be given as

$$G_{\phi,\theta} = R^2 \begin{pmatrix} 1 & 0 \\ 0 & \cos^2\phi \end{pmatrix}.$$

Actually, we can write

$$\begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \underbrace{R \begin{pmatrix} -\cos\theta\sin\phi & -\sin\theta\cos\phi \\ -\sin\theta\sin\phi & \cos\theta\cos\phi \\ \cos\phi & 0 \end{pmatrix}}_{J} \begin{pmatrix} d\phi \\ d\theta \end{pmatrix}.$$

Then, we readily have

$$J^T J = G_{\phi,\theta}.$$

Consider the transformation $\mathcal{T}$, from the plane to the sphere centered at $(\frac{1}{2},\frac{1}{2})$ with radius $R=\frac{1}{2}$, given by

$$\mathcal{T}: \begin{cases} \tilde{\phi} = 2\arctan\left(\sqrt{x^2+y^2}\right) - \dfrac{\pi}{2}, \\ \tilde{\theta} = \arctan\left(\dfrac{y}{x}\right). \end{cases}$$

The corresponding Jacobian is

$$J_{x,y} = \begin{pmatrix} \dfrac{2x}{(1+x^2+y^2)\sqrt{x^2+y^2}} & \dfrac{2y}{(1+x^2+y^2)\sqrt{x^2+y^2}} \\ -\dfrac{y}{x^2+y^2} & \dfrac{x}{x^2+y^2} \end{pmatrix}.$$

We have

$$\begin{aligned} \cos^2\phi &= \cos^2\left(2\arctan\left(\sqrt{x^2+y^2}\right) - \frac{\pi}{2}\right) = \sin^2\left(2\arctan\left(\sqrt{x^2+y^2}\right)\right) \\ &= 4\sin^2\left(\arctan\left(\sqrt{x^2+y^2}\right)\right)\cos^2\left(\arctan\left(\sqrt{x^2+y^2}\right)\right). \end{aligned}$$

Using

$$\cos^2 x = \frac{1}{1+\tan^2 x},$$

we have

$$\cos^2(\arctan(x)) = \frac{1}{1+x^2},$$

and thus

$$\sin^2\left(\arctan\left(\sqrt{x^2+y^2}\right)\right)\cos^2\left(\arctan\left(\sqrt{x^2+y^2}\right)\right) = \frac{x^2+y^2}{(1+x^2+y^2)^2}.$$

It follows that

$$\cos^2\phi = 4\frac{x^2+y^2}{(1+x^2+y^2)^2}.$$

We can write that

$$J_{x,y}^T G_{\phi,\theta} J_{x,y} = \left(\frac{1}{(1+x^2+y^2)}\right)^2 I$$

and conclude that

$$\begin{pmatrix} d\phi \\ d\theta \end{pmatrix}^T G_{\phi,\theta} \begin{pmatrix} d\phi \\ d\theta \end{pmatrix} = \mu^2 \left\| \begin{pmatrix} dx \\ dy \end{pmatrix} \right\|^2,$$

where

$$\mu = \frac{1}{(1+x^2+y^2)}. \tag{6.2}$$

It proves that $\mathcal{T}$ is a conformal transformation with a conformal factor μ.

6.1. Conformal mapping from a sphere onto itself. First, we define the conformal mapping from the sphere onto itself, which is a diffeomorphism. A way to perform the conformal mapping on the sphere is to first apply a stereoscopic transform from the sphere to the plane $\mathbb{C}$ defined by

$$z = \tan\left(\frac{\phi}{2} + \frac{\pi}{4}\right) e^{i\theta}. \tag{6.3}$$

Then, choose $a, b, c \in \mathbb{C}$ that define a conformal Moebius transform on the plane

$$w = \frac{(c-b)(z-a)}{(c-a)(z-b)}. \tag{6.4}$$

Finally, apply an inverse stereoscopic projection

$$\begin{aligned} \tilde{\phi} &= 2\arctan\left(\|w\|\right) - \frac{\pi}{2}, \\ \tilde{\theta} &= \angle(w). \end{aligned} \tag{6.5}$$

The inverse stereoscopic projection is a conformal mapping with a conformal factor defined in (6.2). Its inverse transform, that is, the stereoscopic projection, is also conformal, with a conformal factor $\tilde{\mu} = \frac{1}{\mu}$. It can be shown that the Moebius transform (6.4) is a conformal mapping and that its conformal factor is given by

$$\mu_w = \left\| \frac{(c-b)(b-a)}{(c-a)(z-b)^2} \right\|. \tag{6.6}$$

Combining (6.6) and (6.2), we conclude that the conformal factor of the conformal mapping from the sphere into itself is given by

$$\mu = \left\| \frac{(c-b)(b-a)}{(c-a)(z-b)^2} \right\| \frac{1+\|z\|^2}{1+\|w\|^2}. \tag{6.7}$$

We can provide a geometric interpretation to the conformal parameters (a, b, c). Let

$$\begin{aligned} a &= \tan\left(\frac{\phi_a}{2} + \frac{\pi}{4}\right) e^{i\theta_a}, \\ b &= \tan\left(\frac{\phi_b}{2} + \frac{\pi}{4}\right) e^{i\theta_b}, \\ c &= \tan\left(\frac{\phi_c}{2} + \frac{\pi}{4}\right) e^{i\theta_c}. \end{aligned} \tag{6.8}$$

We notice that $w(a) = 0$, $w(b) = \infty$, and $w(c) = 1$. This means that the point a is sent to the south pole, the point b to the north pole, and the point c is mapped to the equator at location $(\tilde{\phi}_c, \tilde{\theta}_c) = (0, 0)$.

6.2. Rotation using the conformal parameters. The simplest example of conformal mapping from the sphere into itself is the rotation. In a rotation, the angle $\angle(AOB)$ has to be equal to the angle of the corresponding point in the mapping $\angle(\tilde{A}\tilde{O}\tilde{B})$, and the same property holds for the point c. Since a and b are mapped to the poles, they have to be aligned with o, the center of the coordinate system. Then, if a Moebius transform is a rotation, $\phi_b - \phi_a = \pi$ and $\theta_b = \theta_a$. Moreover, the vector $\overrightarrow{OC}$ has to be perpendicular to the vector $\overrightarrow{OA}$. Writing these equations and using the parametrization (6.1), we can show that

$$\cos\theta_a \cos\phi_a \cos\theta_c \cos\phi_c + \sin\theta_a \cos\phi_a \sin\theta_c \cos\phi_c + \sin\phi_a \sin\phi_c = 0$$

or, equivalently,

$$\cos\phi_a \cos\phi_c \cos(\theta_a - \theta_c) + \sin\phi_a \sin\phi_c = 0.$$

We can summarize the conditions on the parameters (a, b, c) that yield a rotation:

$$\phi_b - \phi_a = \pm\pi, \tag{6.9}$$

$$\theta_b = \theta_a, \tag{6.10}$$

$$\cos\phi_a \cos\phi_c \cos(\theta_a - \theta_c) + \sin\phi_a \sin\phi_c = 0. \tag{6.11}$$

Theorem 6.1. *Any conformal mapping from $\mathbb{S}^2$ to $\mathbb{S}^2$ can be realized by controlling the parameters $(\phi_b, \theta_b, \phi_c)$ up to rotation of the sphere. That is, a conformal mapping that sends the point (a, b, c) to the south and north poles, and to the equator as defined by (6.8), can be realized by first setting the parameters $(\phi_b, \theta_b, \phi_c)$ and then rotating the resulting mapping on the sphere.*

Proof. Let us consider the conformal transformation defined by its parameters (a, b, c). We first choose a conformal transformation defined by $(\tilde{a}, \tilde{b}, \tilde{c})$, where $\tilde{a} = a, \theta_{\tilde{c}} = \theta_c$. Let $\tilde{b}$ be defined with $(\tilde{\phi}_b, \tilde{\theta}_b)$ such that b is mapped onto $-a$, and $\tilde{c}$ with $\tilde{\theta}_c = \theta_c$ and $\tilde{\phi}_c$ is such that (6.11) holds for c.

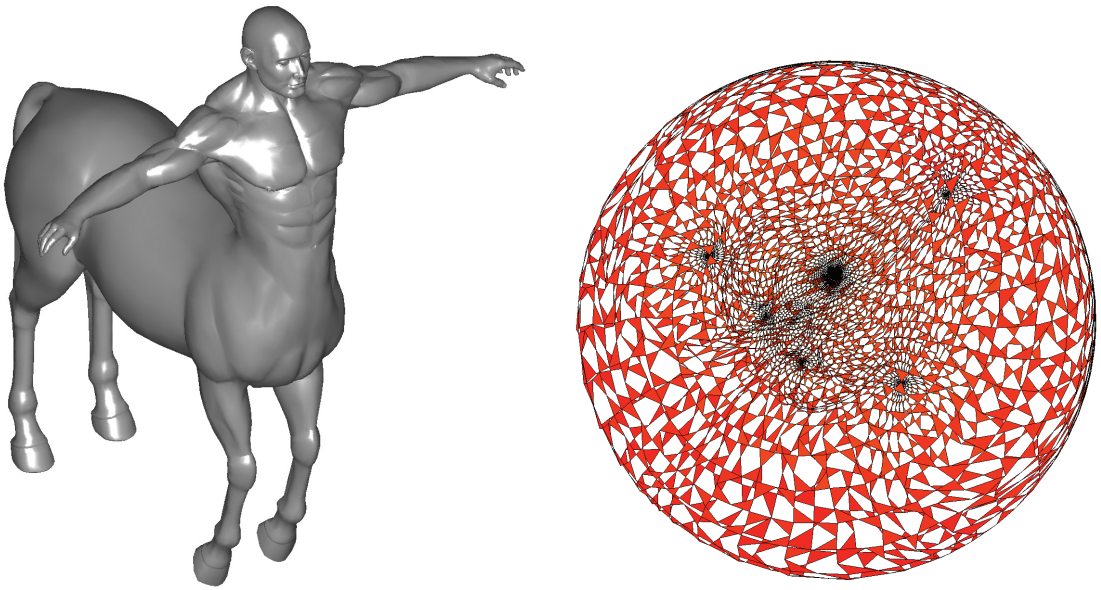


Figure 5. *Centaur and its optimized conformal mapping on a sphere.*

Then, perform the rotation that maps a onto the north pole. $\tilde{b}$ will be mapped onto the south pole, and $\tilde{c}$ will be mapped onto the equator, since (6.9), (6.10), and (6.11) hold. Then we just have to choose the third degree of rotation such that $\tilde{c}$ is mapped onto the point $\phi = 0,\ \theta = 0$.

Then this procedure is a conformal mapping that is equivalent to the mapping defined by (a, b, c). ■

Hence, we can optimize the conformal factor on the sphere using the three parameters $(\phi_b, \theta_b, \phi_c)$. We then solve the following nonconvex optimization problem:

$$\min_{\phi_b,\theta_b,\phi_c} \left[\min_i \left(\mu_i \left\| \frac{(c-b)(b-a)}{(c-a)(z_i-b)^2} \right\| \frac{1+\|z_i\|^2}{1+\|w_i\|^2} \right) \right]$$

s.t.

$$a = \tan\left(\frac{\phi_a}{2} + \frac{\pi}{4}\right) e^{i\theta_a},$$

$$b = \tan\left(\frac{\phi_b}{2} + \frac{\pi}{4}\right) e^{i\theta_b},$$

$$c = \tan\left(\frac{\phi_c}{2} + \frac{\pi}{4}\right) e^{i\theta_c},$$

$$\forall i, \quad w_i = \frac{(c-b)(z-a)}{(c-a)(z_i-b)},$$

$$\forall i, \quad z_i = \tan\left(\frac{\phi_i}{2} + \frac{\pi}{4}\right) e^{i\theta_i}.$$

As an example we obtain the mapping for the centaur shown in Figure 5.

This mapping could also be used to compute geodesic distances on a zero-genus surface using weighted fast marching on a regularly sampled sphere.

7. Bounding the conformal factor. Consider S, a smooth surface embedded in $\mathbb{R}^3$, and G, its induced metric. Let $u : S \to \mathbb{R}$ be a function defined on the surface. We can define another metric $\bar{G} = \mu G$, which is *conformal* to the original one. The Gaussian curvature $\bar{k}$ of the new metric is given by [7]

$$\bar{k} = \frac{1}{\mu}\left(k - \frac{\Delta_G \log(\mu)}{2}\right),$$

where k is the Gaussian curvature on $\{S, G\}$, and Δ_G is the Laplace–Beltrami operator. In the case of conformal mapping to the plane, the target curvature of the new metric is zero. Then the above relation becomes

$$\Delta_G \log(\mu) = 2k.$$

Let us introduce a fundamental property of the Laplace–Beltrami operator.

Definition 7.1. *A linear differential operator L of order n on a domain Ω in $\mathbb{R}^d$ given by*

$$Lf = \sum_{\|\alpha\| \leqslant n} a_\alpha(x) \partial^\alpha f$$

is called elliptic if, for every x in Ω and every nonzero ξ in $\mathbb{R}^d$,

$$\sum_{\|\alpha\|=n} a_\alpha(x) \xi^\alpha \neq 0.$$

Lemma 7.2. *The Laplace–Beltrami operator is an elliptic operator.*

Proof. We have $\Delta_G f = \operatorname{trace}\left(G^{-1}\nabla^2 f\right) + v^t \nabla f$, where $v_j = \frac{1}{\sqrt{g}} \sum_i \partial_i \left(\sqrt{g} g^{ij}\right)$. Then, the Δ_G highest order derivative terms are given by $\operatorname{trace}(G^{-1}\nabla^2 f)$. Taking a vector $\xi \neq 0 \in \mathbb{R}^2$, we have, with the notation of Definition 7.1, $\sum_{\|\alpha\|=2} a_\alpha(x)\xi^\alpha = \operatorname{trace}(\xi^T G^{-1} \xi) \neq 0$ since G^{-1} is a positive definite matrix. This proves that the Laplace–Beltrami operator is elliptic. ∎

The following lemma gives us an upper bound over the conformal factor when the target domain is bounded.

Lemma 7.3. *Given a C^∞ domain $C \in \mathbb{R}^2$, with a metric G, there exists a function b such that for any function $f : C \to \mathbb{R}$ s.t. for all $p \in \partial C : f(p) = 0$, and a positive real number k such that $\|\Delta_G f\| \leqslant k$, we have*

$$\sup_{x \in C} \{\|f(x)\|\} \leqslant b(k).$$

Proof. According to the elliptic regularity theorem, for any $q \in]1, \infty[$, if C is regular, if Δ_G is an elliptic operator, and if $\Delta_G f \in L^q(C)$, then $f \in W^{2,q}(C)$, where $W^{2,q}(C)$ is the $(2, q)$-Sobolev space of C, and there exists a function $g_C^G(q)$ that depends only on C, G, and q such that

$$\|f\|_{W^{2,q}} \leqslant g_C^G(q) \|\Delta_G f\|_{L^q}.$$

Moreover, the Sobolev injection theorem states that if $q > 2$, then there exists a function $h_C^G(q)$ that depends only on C and q such that

$$\|f\|_{C^2(C)} \leqslant h_C^G(q) \|f\|_{W^{2,q}},$$

where $\|f\|_{C^1(C)} = \sup_{x\in C}\{\|f(x)\|\}$. We can then conclude that

$$\sup_{x\in C}\{\|f(x)\|\} \leqslant h_C^G(q) g_C^G(q)\mu(C)k = b(k). \qquad \blacksquare$$

Using the relation $u = \frac{1}{2}\log\mu$, we can choose the conformal factor such that $\mu = 1$ on ∂C. Lemma 7.3 states that $|\log\mu|$ is upper bounded, which proves that μ is lower and upper bounded, and that

$$\frac{\sup|\mu|}{\inf|\mu|} \leqslant e^{2b(k)}.$$

This bound justifies using the conformal map for numerically computing geometric measures such as geodesic distances. We can then conclude that since it is possible to find a boundary condition for the conformal factor that leads to a global upper bound over the ratio, optimization over the conformal factor is justified. The computation of geometric quantities in the conformal mapping in the case of bounded ratio is thereby consistent.

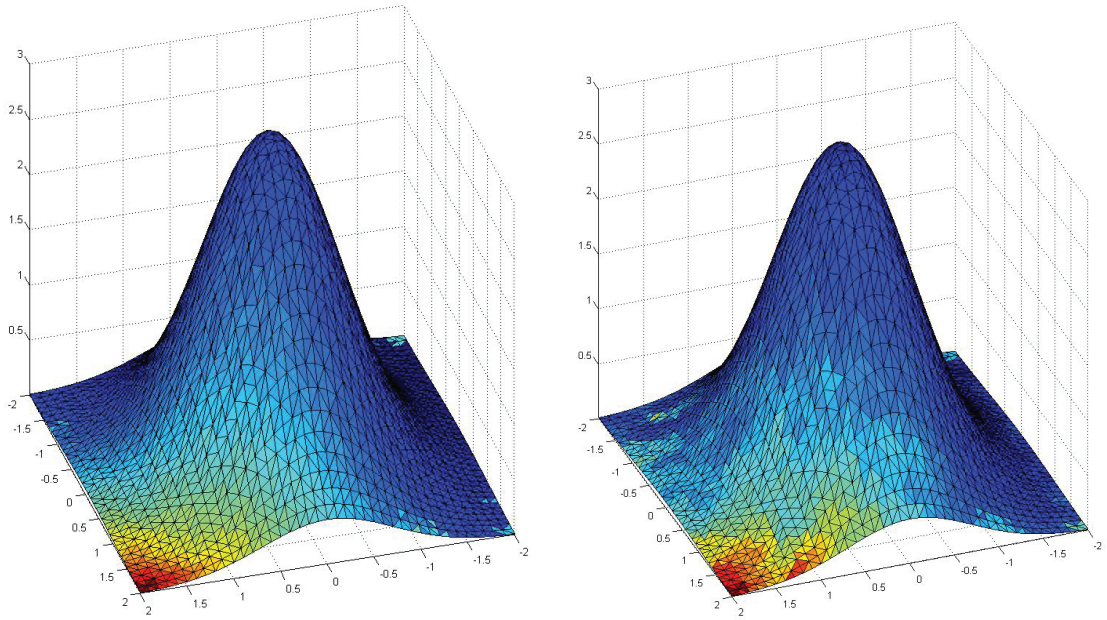
8. Anisotropic metric reconstruction. In the example in section 8.1, we deal with an anisotropic metric reconstruction problem. Some sources are located at r_s and some sensors at l_t. The sensors provide the traveling time of a signal from r_s to l_t that corresponds to the weighted geodesic distance, which we denote by $d(r_s, l_t)$. Our goal is to recover the *smooth* metric $\xi(z)$ from the traveling time information. In Devir et al. [5] it was first shown how to reconstruct an unknown surface from a given metric taking the metric derivative. The other idea proposed by Benmansour et al. [1] in the case of an isotropic inhomogeneous metric consists of expressing this problem as an optimization of $\min_\xi \sum_{s,t}(d_\xi(r_s,l_t) - d(r_s,l_t))^2 + \mu\|\nabla\xi\|^2$, where the term $\|\nabla\xi\|^2$ indicates the assumption that the unknown metric is smooth. We can use a steepest descent method to find the minimum, where the optimization variable is ξ. Thus, we need to be able to compute $\nabla_\xi d_\xi(r_s, l_t)$. This is equivalent to finding the derivative $\nabla_\xi U^{r_s}(l_t)$, where $U^{r_s}(l_t)$ denotes the geodesic distance from r_s to l_t. In our case, the metric is not flat. The problem of using the FMM [3] on flat domains is that the update step is not as simple as in the case of a flat metric. Our idea is to first conformally flatten the problem by mapping our manifold to $\mathbb{R}^2$ and then running the two-dimensional metric recovering algorithm computation [1] on the conformal plane.

8.1. Metric reconstruction on the manifold. When mapping a manifold using a conformal map, the local metric is multiplied by a conformal factor. A simple reconstruction algorithm reads as follows:

1. Map the surface S into $\mathbb{R}^2$ and obtain the conformal factor μ for each point.
2. Run the flat metric reconstruction algorithm defined in [1] and recover the local flat metric $\tilde{\xi}$.
3. Recover the original metric $\xi = \frac{\tilde{\xi}}{\mu}$.

8.2. Simulation. We ran the three-dimensional metric reconstruction on the surface in Figure 6 where the local metric is represented by a color code, using six sources and six sensors. The reconstructed metric is presented in Figure 6.

The mean average squared of the ratio error $\frac{1}{\text{Area}}\int \|\frac{\tilde{\xi}_{\text{real}} - \tilde{\xi}_{\text{recovered}}}{\tilde{\xi}_{\text{real}}}\| da$ is 0.06.



(a) Original metric (b) Recovered metric

Figure 6. *Original and recovered metric.*

9. Conclusions. Conformal mapping of a surface onto a plane is a powerful analysis procedure. Still, in order to justify its usage as a computational tool one needs to control the numerical behavior of such a mapping. We proved that a lower bound over the ratio between the minimal and the maximal conformal factor exists. We demonstrated that this theoretical bound does not help much in practice. Next, we formulated optimization problems that maximize this ratio. It allowed us to efficiently and accurately compute geodesic distances using regular sampling of the plain.

Appendix. Introduction to fast marching method.

A.1. General background. Let us consider a two-dimensional surface X embedded in $\mathbb{R}^3$. This surface is associated with a metric G. We denote by $z = (u, v)$ the coordinates of a point in $\mathbb{R}^2$ and by x the coordinates of a point in the embedded space X. Our motivation is to measure the geodesic distance between two points, which is defined by the path minimal length among all curves lying on X that connect these two points. Formally, we introduce the function $U : X^2 \to \mathbb{R}^+$ that represents the geodesic distance between $x(u, v)$ and x_0 in the manifold, defined by

$$U(u,v) = \min_{\substack{\gamma \in X \\ \text{s.t} \\ \gamma(0)=x_0 \\ \gamma(1)=x(u,v)}} \int_0^1 \sqrt{\dot{\gamma}(t)^T G(\gamma(t))\dot{\gamma}(t)}dt. \tag{A.1}$$

Geodesic distance computation is important in shape processing and especially shape comparisons since it is preserved under isometric transformations. Dijkstra's algorithm is a graph

search algorithm that solves the single-source shortest path problem for a graph with non-negative costs along the edges, producing a shortest path tree. The FMM is a generalization of Dijkstra's algorithm for a two-dimensional manifold equipped with a Riemannian metric [18, 22, 23, 20, 11, 15]. The FMM differs from Dijkstra's algorithm by its update step.

A.2. Eikonal equation. In order to compute U, we notice that (A.1) describes the length of the minimal path between two points. Consider a point z_0 and define $U(z) = U(z_0, z)$. Considering a geodesic circle of radius δ in the neighborhood of z, we can assume that the minimal path between z_0 and z passes through this circle. The distance from any point on the circle to z is δ by definition. Therefore, the location of the point on the circle $z - \delta\vec{k}$ has to be chosen to minimize its distance from z_0. That is,

$$\begin{aligned} U(z) &= \min_{\substack{\vec{k}\in\mathbb{R}^2 \\ \vec{k}^T G(z)\vec{k}=1}} \left\{ U\left(z - \delta\vec{k}\right) + \delta + O(\delta^2) \right\} \\ &= \min_{\substack{\vec{k}\in\mathbb{R}^2 \\ \vec{k}^T G(z)\vec{k}=1}} \left\{ U(z) - \delta\nabla_z U_z(z)\vec{k} + \delta + O(\delta^2) \right\}. \end{aligned}$$

It follows that

$$\min_{\substack{\vec{k}\in\mathbb{R}^2 \\ \vec{k}^T G\vec{k}=1}} \{-\nabla_z U_z(z)\vec{k} + 1\} = 0$$

can be written as $\nabla_z U_z(z)\vec{\tilde{k}} = 1$, where

$$\vec{\tilde{k}} = \operatorname*{argmax}_{\substack{\vec{k}\in\mathbb{R}^2 \\ \vec{k}^T G\vec{k}=1}} \{\nabla U(z)\vec{k}\}.$$

Note that there exists a matrix J such that $G = J^T J$. Define $\vec{\tilde{k}} = J\vec{k}$ such that we have $(J^T J)^{-1} J^T \vec{\tilde{k}} = \vec{k}$. The problem can be rewritten as

$$\vec{\tilde{k}} = \operatorname*{argmax}_{\substack{\vec{k}\in\mathbb{R}^2 \\ \|\vec{k}\|=1}} \{\nabla_z U_z(z)(G^{-1})J^T\vec{k}\}.$$

The maximum is realized for $\vec{k} = \frac{J(G^{-1})(\nabla_z U_z(z))^T}{\sqrt{\nabla_z U_z(z)(G^{-1})J^T J(G^{-1})(\nabla_z U_z(z))^T}}$. Hence, the equation becomes $\left|\nabla_z U_z(z)(G^{-1})(\nabla_z U_z(z))^T\right| = 1$. When $G = \mathbf{1}$, we have $\|\nabla_z U(z)\| = 1$, the known eikonal equation in isotropic homogeneous domains.

A.3. Fast marching for geodesic distance computation.

A.3.1. Description of the algorithm. We establish the relation between $\nabla_z U(z)$ and G. We can use it to compute the geodesic distance between a point z_0 and z with the *fast marching method*. The algorithm operates on a rectangular grid. Its update protocol is similar to Djikstra's algorithm and can be described as follows.

Algorithm 2. Fast marching.

Require: i_{z_0}, j_{z_0} $\{z_0$ is located on the grid at $(i_{z_0}, j_{z_0})\}$
  for all $(i,j) \in \Omega$ **do**
    $U(i,j) \leftarrow \infty$
    $s(i,j) \leftarrow$ 'Known'
  end for
  $s_{i_{z_0},j_{z_0}} \leftarrow$ 'Far'
  $U(i_{z_0}, j_{z_0}) \leftarrow$ '0'
  while $\exists (i,j) | s_{i,j} ==$ 'Trial' **do**
    Find (i,j) such that $s_{i,j} ==$ 'Trial' with the minimal value of $U(i,j)$,
    $s_{i,j} \leftarrow$ 'Known'
    for all $(i',j') \in \mathcal{N}(i,j)$ such that $s_{i',j'} \neq$ 'Known' **do**
      $s_{i',j'} \leftarrow$ 'Trial'
      Update the value of $U(i',j')$
    end for
  end while

This algorithm allows us to compute U for all the grid points.

We use the Rouy and Tourin [17] numerical approximation of the eikonal equation $|\nabla u| = \mu$. On a uniform regular grid, $U(i,j)$ denotes $u(ih, jh)$, where h stands for the sampling interval along the x and y directions. Then the numerical viscosity solution can be derived by approximating

$$|u_x| \approx \max\left(0, U(i,j) - \min(U(i+1,j), U(i-1,j))\right)$$

and

$$|u_y| \approx \max\left(0, U(i,j) - \min(U(i,j+1), U(i,j-1))\right).$$

That leads to a quadratic equation that has to be evaluated for the solution of $U(i,j)$ as the update step in our scheme. The approximation of the eikonal equation can be found in [10, 19, 11, 18]. The resulting solution of a quadratic equation takes into consideration the values of all neighboring points and their geometric arrangement.

REFERENCES

[1] F. Benmansour, G. Carlier, G. Peyré, and F. Santambrogio, *Derivatives with respect to metrics and applications: Subgradient marching algorithm*, Numer. Math., 116 (2010), pp. 357–381.

[2] A. Ben-Tal and M. Zibulevsky, *Penalty/barrier multiplier methods for convex programming problems*, SIAM J. Optim., 7 (1997), pp. 347–366.

[3] A. Bronstein, M. Bronstein, and R. Kimmel, *Weighted distance maps computation on parametric three-dimensional manifolds*, J. Comput. Phys., 225 (2006), pp. 771–784.

[4] A. Bronstein, M. Bronstein, and R. Kimmel, *Numerical Geometry of Non-Rigid Shapes*, Springer, New York, 2008.

[5] Y. S. Devir, G. Rosman, A. M. Bronstein, M. M. Bronstein, and R. Kimmel, *On reconstruction of non-rigid shapes with intrinsic regularization*, in Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), 2009, pp. 272–279.

[6] C. Frederick and E. L. Schwartz, *Conformal image warping*, IEEE Comput. Graph. Appl., 10 (1990), pp. 54–61.
[7] X. Gu, S. Wang, J. Kim, Y. Zeng, Y. Wang, H. Qin, and D. Samaras, *Ricci flow for 3D shape analysis*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV '07), 2007, pp. 1–8.
[8] X. Gu, Y. Wang, T. F. Chan, P. M. Thompson, and S.-T. Yau, *Genus zero surface conformal mapping and its application to brain surface mapping*, IEEE Trans. Med. Imag., 23 (2004), pp. 949–957.
[9] M. Jin, Y. Wang, S.-T. Yau, and X. Gu, *Optimal global conformal surface parameterization*, in Proceedings of the Conference on Visualization '04 (VIS '04), IEEE Computer Society, Washington, DC, 2004, pp. 267–274.
[10] R. Kimmel, *Numerical Geometry of Images: Theory, Algorithms, and Applications*, Springer-Verlag, New York, 2004.
[11] R. Kimmel and J. A. Sethian, *Fast marching methods on triangulated domains*, Proc. Natl. Acad. Sci. USA, 95 (1998), pp. 8341–8435.
[12] R. Kimmel and J. A. Sethian, *Optimal algorithm for shape from shading and path planning*, J. Math. Imaging Vision, 14 (2001), pp. 237–244.
[13] Y. Lipman and I. Daubechies, *Conformal Wasserstein distances: Comparing surfaces in polynomial time*, Adv. Math., 227 (2011), pp. 1047–1077.
[14] Y. Lipman and T. Funkhouser, *Möbius voting for surface correspondence*, ACM Trans. Graph., 28 (2009), 72.
[15] F. Mémoli and G. Sapiro, *Fast computation of weighted distance functions and geodesics on implicit hyper-surfaces*, J. Comput. Phys., 173 (2001), pp. 730–764.
[16] K. Polthier, *Conjugate Harmonic Maps and Mimimal Surfaces*, Preprint 446, SFB 288, TU-Berlin, Berlin, Germany, 2000.
[17] E. Rouy and A. Tourin, *A viscosity solutions approach to shape-from-shading*, SIAM J. Numer. Anal., 29 (1992), pp. 867–884.
[18] J. A. Sethian, *A fast marching level set method for monotonically advancing fronts*, Proc. Natl. Acad. Sci. USA, 93 (1996), pp. 1591–1595.
[19] J. A. Sethian, *Fast marching methods*, SIAM Rev., 41 (1999), pp. 199–235.
[20] A. Spira and R. Kimmel, *An efficient solution to the eikonal equation on parametric manifolds*, Interfaces Free Bound., 6 (2004), pp. 315–327.
[21] V. Surazhsky and T. Surazhsky, *Fast exact and approximate geodesics on meshes*, ACM Trans. Graph., 24 (2005), pp. 553–560.
[22] J. N. Tsitsiklis, *Efficient algorithms for globally optimal trajectories*, IEEE Trans. Automat. Control, 40 (1995), pp. 1528–1538.
[23] O. Weber, Y. S. Devir, A. Bronstein, M. Bronstein, and R. Kimmel, *Parallel algorithms for approximation of distance maps on parametric surfaces*, ACM Trans. Graph., 27 (2008), 104.
[24] W. Zeng, L. M. Lui, F. Luo, T. F. Chan, S.-T. Yau, and D. X. Gu, *Computing quasiconformal maps using an auxiliary metric and discrete curvature flow*, Numer. Math., 121 (2012), pp. 671–703.
[25] M. Zibulevsky, *PBM Toolbox for Constrained Nonlinear and Semidefinite Optimization (in MATLAB)*, http://ie.technion.ac.il/~mcib/pbm.html.