SIAM J. IMAGING SCIENCES
Vol. 17, No. 4, pp. 2110–2141

Spherical Density-Equalizing Map for Genus-0 Closed Surfaces*

Zhiyuan Lyu†, Lok Ming Lui†, and Gary P. T. Choi†

Abstract. Density-equalizing maps are a class of mapping methods in which the shape deformation is driven by prescribed density information. In recent years, they have been widely used for data visualization on planar domains and planar parameterization of open surfaces. However, the theory and computation of density-equalizing maps for closed surfaces are much less explored. In this work, we develop a novel method for computing spherical density-equalizing maps for genus-0 closed surfaces. Specifically, we first compute a conformal parameterization of the given genus-0 closed surface onto the unit sphere. Then we perform density equalization on the spherical domain based on the given density information to achieve a spherical density-equalizing map. The bijectivity of the mapping is guaranteed by introducing an overlap correction scheme based on quasi-conformal theory throughout the density-equalizing iterative process. We further propose a method for incorporating the harmonic energy and landmark constraints into our formulation to achieve landmark-aligned spherical density-equalizing maps balancing different distortion measures. Using the proposed methods, a large variety of spherical parameterizations can be achieved. Applications to surface registration, remeshing, and data visualization are presented to demonstrate the effectiveness of our methods.

Key words. surface parameterization, density-equalizing map, quasi-conformal theory, genus-0 closed surfaces, diffusion

MSC codes. 65D18, 68U05, 30C70, 76R50

DOI. 10.1137/24M1633911

1. Introduction.

Surface parameterization is the process of mapping a complicated surface onto a simple standardized domain. For genus-0 closed surfaces, it is common to consider parameterizing them onto the unit sphere. There has been a vast number of works on the computation of spherical conformal parameterizations [3, 5, 6, 10, 11, 18, 23, 41], which preserve angles under the mappings. There have also been some works on the computation of spherical area-preserving parameterization [2, 12, 34, 35, 40, 42]. Some other methods have considered the problem of achieving a balance between various distortion measures for spherical parameterization [7, 25, 26, 29, 38, 39].

Density-equalizing maps [15] is a widely used approach for producing cartograms, for which a planar map is deformed based on the physical principle of density diffusion. Under a density-equalizing map, regions with a larger prescribed quantity (called the *population*) expand and those with a smaller population shrink, so that the area ratio of different regions

*Received by the editors January 24, 2024; accepted for publication (in revised form) August 2, 2024; published electronically October 18, 2024.
https://doi.org/10.1137/24M1633911
Funding: The second author was supported by HKRGC GRF (Project ID 14306721). The third author was supported by the CUHK Vice-Chancellor Early Career Professorship Research Startup Fund (Project Code 4933624) and the Croucher Foundation Start-up Allowance.
†Department of Mathematics, The Chinese University of Hong Kong, Hong Kong (zylyu@math.cuhk.edu.hk, lmlui@math.cuhk.edu.hk, ptchoi@cuhk.edu.hk).

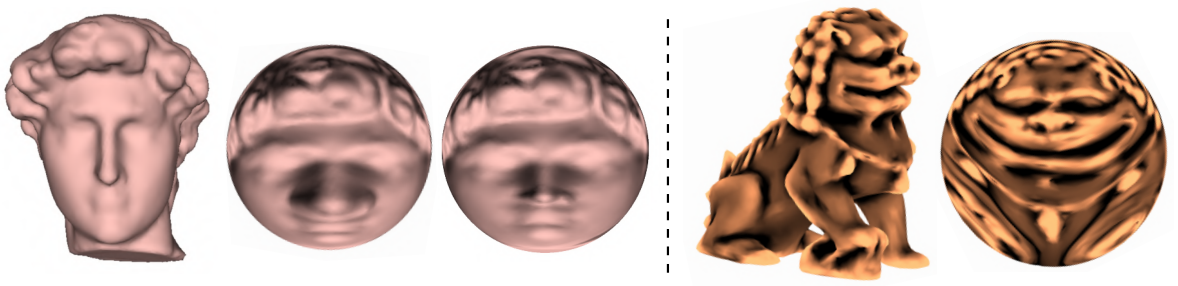

Figure 1. Examples of the spherical density-equalizing maps obtained by our SDEM method. *(Left) The David model and two spherical density-equalizing maps obtained by our SDEM method, with the nose region enlarged or shrunk based on the prescribed population. (Right) The Chinese Lion model and the spherical area-preserving parameterization obtained by our SDEM method.*

in the resulting deformed map reflects the ratio of the input population. In recent years, this method has been extensively applied for the visualization of sociological and biological data [13, 17, 28, 30, 33, 36, 37]. Some recent efforts have been made on improving the computation of density-equalizing maps on 2D or 3D grids [9, 14, 16, 21, 22]. More recently, several surface parameterization methods for open surfaces have been proposed using the idea of density-equalizing maps [4, 8, 27]. However, since genus-0 closed surfaces are topologically equivalent to the sphere but not the plane, the existing planar mapping formulations are not suitable for handling such surfaces. While it is possible to use different projection methods to map planar results onto the sphere, such projections will unavoidably introduce distortions in area (e.g., the stereographic projection), angle (e.g., the Lambert equal-area projection), or both (e.g., the Mercator projection), thereby leading to inaccuracies in the final spherical density-equalizing maps. It is more natural to skip the planar parameterization and directly compute density-equalizing maps on the sphere. Also, one may want to combine density equalization and other surface mapping criteria to achieve different desired spherical mapping effects.

In this work, we first develop a novel method for computing bijective spherical density-equalizing maps (abbreviated as SDEM) for genus-0 closed surfaces (see Figure 1 for examples) in which the bijectivity of the mappings is ensured by introducing an overlap correction scheme based on quasi-conformal theory throughout the density-equalizing iterative process. Then we propose an energy minimization model combining density-equalization, conformality, and landmark-matching conditions for producing landmark-aligned spherical density-equalizing maps (abbreviated as LSDEM) that balance between different distortion measures. We apply our proposed methods for different mapping problems and practical applications to demonstrate their effectiveness.

The rest of this paper is organized as follows. In section 2, we review the mathematical background of this work. In section 3, we describe our two proposed methods for spherical density-equalizing maps and landmark-aligned spherical density-equalizing maps in detail. In section 4, we present numerical experiments to assess the performance of our proposed methods. In section 5, we introduce the applications of our methods to surface registration, surface remeshing, and spherical data visualization. We conclude our work and discuss possible future directions in section 6.

2. Mathematical background.

In this section, we describe some basic theories of density-equalizing maps and quasi-conformal geometry.

2.1. Density-equalizing maps.

Gastner and Newman [15] proposed a method for computing *density-equalizing maps* based on the diffusion process. Given a 2D planar domain, a positive density function ρ is first defined on every unit area of it. The mapping method aims to deform the domain according to ρ to obtain a density-equalizing domain. Moreover, during the deformation process, the regions with high density will be enlarged and the regions with low density will be shrunk. The advection equation is given by

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot \mathbf{j}, \tag{2.1}$$

where $\mathbf{j} = -\nabla\rho$ is the flux by Fick's law of diffusion. Hence, the diffusion equation can be obtained by

$$\frac{\partial \rho}{\partial t} = \Delta\rho. \tag{2.2}$$

The expression of the velocity field in terms of density is

$$\mathbf{v} = -\frac{\nabla\rho}{\rho}. \tag{2.3}$$

Combining the above formulas, the cumulative displacement $\mathbf{r}(t)$ of any point on the map at time t can be calculated by integrating the following velocity field:

$$\mathbf{r}(t) = \mathbf{r}(0) + \int_0^t \mathbf{v}(\mathbf{r}, \tau)\mathrm{d}\tau. \tag{2.4}$$

Note that the diffusion flow is always directed from high-density regions to low-density regions. In the limit $t \to \infty$, the density ρ induced by the above displacement $\mathbf{r}(t)$ will be fully equalized per unit area. An illustration is given in Figure 2.

The density-equalizing mapping method has been widely applied to cartogram creation and sociological data visualization. In recent years, Choi and Rycroft (see [4, 8]) proposed

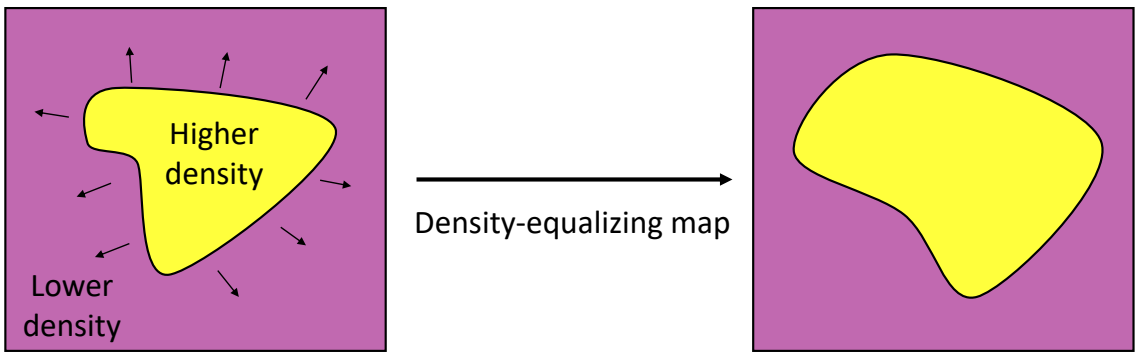

Figure 2. An illustration of density-equalizing maps. *During the diffusion process, the regions with high density will be enlarged and the regions with low density will be shrunk.*

methods for computing the density-equalizing maps for general simply connected open surfaces in $\mathbb{R}^3$. More recently, Lyu, Choi, and Lui [27] developed a method for computing density-equalizing quasi-conformal (DEQ) maps to achieve a balance between density equalization and quasi conformality for simply connected and multiply connected open surfaces.

2.2. Quasi-conformal theory. It is well known that conformal maps preserve angles. *Quasi-conformal maps,* as a generalization of conformal maps, relax the condition by allowing angle distortion within certain bounds. Mathematically, a mapping $f : \overline{\mathbb{C}} \to \overline{\mathbb{C}}$ is called a quasi-conformal map if it satisfies the Beltrami equation

$$\frac{\partial f}{\partial \bar{z}} = \mu(z)\frac{\partial f}{\partial z} \tag{2.5}$$

for some complex-valued function μ with $\|\mu\|_\infty < 1$. μ is called the *Beltrami coefficient* of f, which measures the conformality distortion of the mapping f. In particular, if $\mu = 0$, then (2.5) becomes the Cauchy–Riemann equation, and hence the mapping f is conformal. Intuitively, around a point $z_0 \in \mathbb{C}$, the first-order approximation of f can be expressed as

$$f(z) \approx f(z_0) + f_z(z_0)(z - z_0) + f_{\bar{z}}(z_0)\overline{(z - z_0)} = f(z_0) + f_z(z_0)(z - z_0 + \mu(z_0)\overline{(z - z_0)}). \tag{2.6}$$

The above formula suggests that f maps an infinitesimal circle centered at z_0 to an infinitesimal ellipse centered at $f(z_0)$. Additionally, we can determine the angles at which the magnification and shrinkage are maximized, as well as quantify the degree of magnification and shrinkage at those angles (see Figure 3). More specifically, the angle of maximal magnification is $\arg(\mu(z_0))/2$ with the magnifying factor $|f_z(z_0)|(1 + |\mu(z_0)|)$, and the angle of maximal shrinkage is $(\arg(\mu(z_0)) + \pi)/2$ with the shrinking factor $|f_z(z_0)(|1 - |\mu(z_0)|)$. The maximal dilation of f is $K(f) = \frac{1+\|\mu\|_\infty}{1-\|\mu\|_\infty}$. Thus, the Beltrami coefficient encodes important geometric information about the quasi-conformal map f.

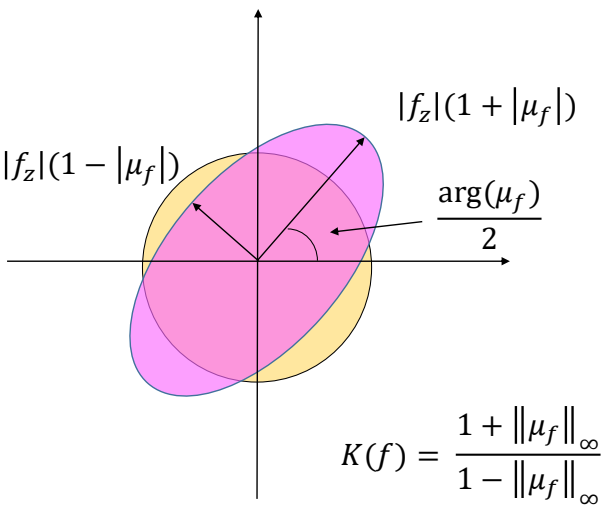

Figure 3. An illustration of quasi-conformal maps. *Under a quasi-conformal map f, an infinitesimal circle is mapped to an infinitesimal ellipse with bounded eccentricity. The maximal magnification, maximal shrinkage, and maximal dilation are all related to the Beltrami coefficient μ.*

Meanwhile, the Beltrami coefficient is closely related to the bijectivity of the quasi-conformal map. More specifically, we have the following result [1, 20].

Theorem 2.1. *If f is a C^1 mapping satisfying $\|\mu_f\|_\infty < 1$, then f is bijective.*

Proof. For a quasi-conformal map $f = u + iv$, the Jacobian matrix J_f is given by

$$J_f = u_x v_y - v_x u_y = |f_z|^2(1 - |\mu|^2). \tag{2.7}$$

Thus, if $\|\mu\|_\infty < 1$ and $|\frac{\partial f}{\partial z}| \neq 0$, then $J_f > 0$ everywhere. Therefore, f is bijective. ∎

Moreover, the Beltrami coefficient of a composition of two quasi-conformal maps can be expressed in the following way. Let $f, g : \overline{\mathbb{C}} \to \overline{\mathbb{C}}$ be two quasi-conformal maps with Beltrami coefficients μ_f and μ_g, respectively. The Beltrami coefficient of the composition map $g \circ f$ is given by

$$\mu_{g\circ f} = \frac{\mu_f + (\mu_g \circ f)\tau}{1 + \overline{\mu_f}(\mu_g \circ f)\tau}, \tag{2.8}$$

where $\tau = \overline{f_z}/f_z$. In particular, if g is a conformal map, then $\mu_{g\circ f} = \mu_f$. In other words, composing a conformal map with a given quasi-conformal map will not change its Beltrami coefficient. This observation holds significant importance in our proposed algorithms, as discussed in the following section.

3. Proposed methods.

This section presents our main proposed methods in this work. Below, we first develop a novel algorithm for computing bijective spherical density-equalizing maps (abbreviated as SDEM) of genus-0 closed surfaces onto a unit sphere. Then we propose another method for computing bijective landmark-aligned spherical density-equalizing maps (abbreviated as LSDEM) of genus-0 closed surfaces.

3.1. Bijective spherical density-equalizing map (SDEM).

Let $\mathcal{M}$ be a genus-0 closed surface represented as a triangular mesh $(\mathcal{V}, \mathcal{E}, \mathcal{F})$, where $\mathcal{V}$ is the set of all vertices, $\mathcal{E}$ is the set of all edges, and $\mathcal{F}$ is the set of all triangular faces. Our goal is to compute a bijective spherical density-equalizing map $f : \mathcal{M} \to \mathbb{S}^2$ based on a prescribed population encoding the desired area changes.

3.1.1. Initial spherical parameterization.

First, we apply the FLASH method proposed by Choi, Lam, and Lui [10] to compute an initial spherical conformal parameterization. FLASH is a north pole–south pole iterative scheme specifically designed to reduce the conformality distortion of the spherical conformal map using quasi-conformal theory. The process begins by selecting a triangle element from $\mathcal{F}$ as the north pole. Next, the method solves the Laplace equation on a triangular domain on the complex plane $\mathbb{C}$ to obtain a conformal map of the input surface with the selected triangle punctured. By employing the inverse stereographic projection, the planar region can then be mapped onto the sphere $\mathbb{S}^2$, and then the punctured element can be added back to the surface to get a spherical parameterization. However, the spherical mapping result often exhibits significant angle distortions near the north pole. To address this issue, the method projects the sphere onto the plane using the south-pole stereographic projection and composes the map with another quasi-conformal map to correct the conformality distortion. Finally, the updated map is projected back onto the sphere to yield the desired spherical conformal parameterization. Readers are referred to [10] for more details.

One common issue of global conformal parameterizations is that the area may be largely distorted. In order to mitigate this problem, an additional step of composing the spherical conformal parameterization with a Möbius transformation can be introduced, as discussed in [6]. Specifically, one can start by projecting the afore-mentioned spherical conformal parameterization result onto $\overline{\mathbb{C}}$ using the stereographic projection and then apply a Möbius transformation followed by the inverse stereographic projection back onto the sphere. By finding an optimal Möbius transformation, the area distortion of the spherical parameterization can be reduced (see [6] for details). Moreover, as both the stereographic projection and the Möbius transformations are conformal, the resulting updated spherical parameterization is also conformal.

We denote the spherical conformal parameterization obtained using the above-mentioned procedures as $f_0 : \mathcal{M} \to \mathbb{S}^2$.

3.1.2. Density-equalizing map on the sphere. Let $\mathbf{x}$ be a vertex on the sphere $\mathbb{S}^2$ with a positive quantity (the population) prescribed at it, giving a density ρ on $\mathbb{S}^2$ defined as population per unit area. Below we describe our proposed method for achieving density-equalizing maps on the sphere. Note that the density diffusion is a time-dependent process, and so here we denote the initial position of $\mathbf{x}$ as $\mathbf{x}(0)$. At time t, the updated position of the vertex is denoted as $\mathbf{x}(t)$.

Considering the diffusion of ρ, we have the following diffusion equation:

$$\frac{\partial \rho}{\partial t} = \Delta \rho, \tag{3.1}$$

where Δ is the Laplace–Beltrami operator. The density gradient induces a velocity field for the following vertices (Figure 4(a)):

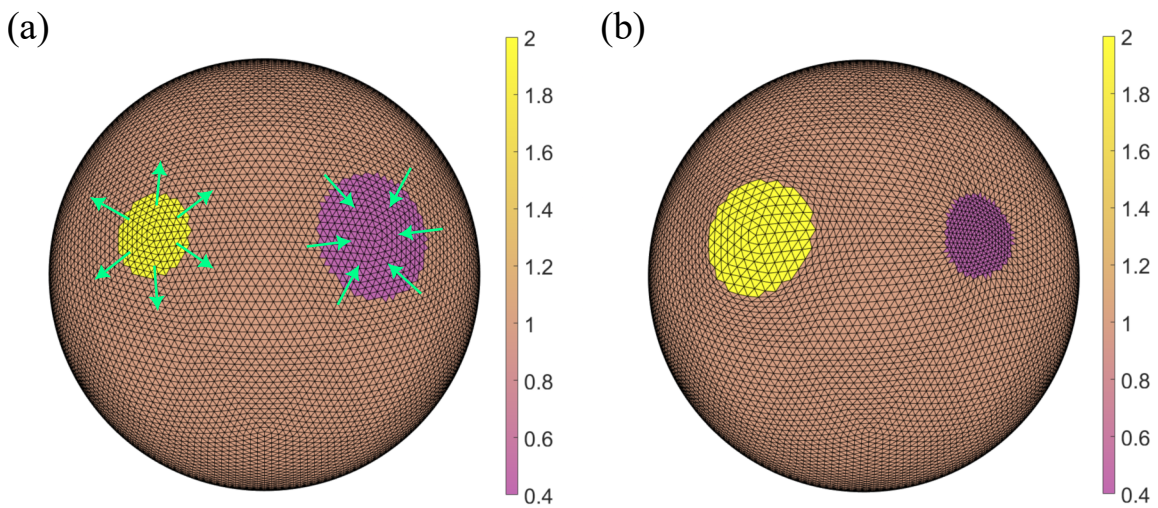

Figure 4. Density-equalizing map on a sphere. (a) *A spherical surface color-coded with a prescribed density ρ. The diffusion of ρ induces a density flux, thereby giving a velocity field on the sphere, as indicated by the green arrows.* (b) *Under the spherical density-equalizing map, the region with a higher density (in yellow) expands and the region with a lower density (in purple) shrinks.*

$$\mathbf{v}(\mathbf{x}(t),t) = -\frac{\nabla\rho(\mathbf{x}(t),t)}{\rho(\mathbf{x}(t),t)}. \tag{3.2}$$

Note that the velocity field is determined by the prescribed populations and automatically lies on the tangent space of the sphere. Now every vertex on the sphere is displaced according to the following equation:

$$\mathbf{x}(t) = \mathbf{x}(0) + \int_0^t \mathbf{v}(\mathbf{x}(\tau),\tau)d\tau. \tag{3.3}$$

As $t \to \infty$, we obtain a spherical density-equalizing map with the final position of all vertices being $\mathbf{x}(\infty)$ (Figure 4(b)). However, note that in the discrete case, the velocity field may not lie on the tangent space of the sphere. Therefore, we need to add a projection step to obtain a modified velocity field $\widetilde{\mathbf{v}}$ to ensure all the vertices after deformation are still on the sphere, which will be discussed below.

In the discrete case, we begin with the initial density $\rho_{\mathcal{F}}^0 = \frac{\text{Population}}{\text{Triangle Area}}$ defined on each triangle element of the spherical mesh. The diffusion-based deformation is obtained by iteratively updating the positions of all vertices $\{\mathbf{x}_i(t_n)\}_i$ at time $t = t_n$, where $i = 1, 2, \ldots, |\mathcal{V}|$ and $n = 0, 1, \ldots$ with a uniform timestep $\delta t = t_{n+1} - t_n$. Denote $\mathbf{v}_{\mathcal{V}}^n$ as a $|\mathcal{V}| \times 3$ matrix representing the velocity field at all vertices $\{\mathbf{x}_i(t_n)\}_i$ at time $t = t_n$. We also denote $\rho_{\mathcal{V}}^n$ and $\rho_{\mathcal{F}}^n$, respectively, as a $|\mathcal{V}| \times 1$ column vector and a $|\mathcal{F}| \times 1$ column vector representing the density at every vertex or every face at time $t = t_n$. As described in [8], $\rho_{\mathcal{F}}^n$ and $\rho_{\mathcal{V}}^n$ are related by a simple formula:

$$\rho_{\mathcal{V}}^n = M\rho_{\mathcal{F}}^n, \tag{3.4}$$

where M is a $|\mathcal{V}| \times |\mathcal{F}|$ face-to-vertex conversion matrix with

$$M_{ij} = \begin{cases} \frac{\text{Area}(T_j)}{\sum_{T \in N^{\mathcal{F}}(\mathbf{x}_i)} \text{Area}(T)} & \text{if } T_j \text{ is incident to } \mathbf{x}_i, \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

Here $N^{\mathcal{F}}(\mathbf{x}_i)$ represents the set of all triangular faces incident to the vertex $\mathbf{x}_i$. Alternatively, one may also directly prescribe a vertex-based density $\rho_{\mathcal{V}}^n$ and skip (3.4).

To solve (3.1), we discretize the Laplace–Beltrami operator at time $t = t_n$ as

$$\Delta_n = -A_n^{-1}L_n, \tag{3.6}$$

where A_n is a $|\mathcal{V}| \times |\mathcal{V}|$ diagonal matrix (called the lumped mass matrix) with

$$(A_n)_{ii} = \frac{1}{3} \sum_{[\mathbf{x}_i,\mathbf{x}_j,\mathbf{x}_k] \in N^{\mathcal{F}}(\mathbf{x}_i)} \text{Area}([\mathbf{x}_i(t_n), \mathbf{x}_j(t_n), \mathbf{x}_k(t_n)]), \tag{3.7}$$

where $[\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k]$ is a triangle in $N^{\mathcal{F}}(\mathbf{x}_i)$. In other words, each diagonal entry of A_n is the area sum of all triangles surrounding a vertex at time $t = t_n$. L_n is the $|\mathcal{V}| \times |\mathcal{V}|$ mesh Laplacian matrix given by the following cotangent formula [32]:

$$(L_n)_{ij} = \begin{cases} -\frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij}) & \text{if } [\mathbf{x}_i, \mathbf{x}_j] \in \mathcal{E}, \\ -\sum_{k \neq i} (L_n)_{ik} & \text{if } j = i, \\ 0 & \text{otherwise,} \end{cases} \tag{3.8}$$

where α_{ij} and β_{ij} are the two angles opposite to the edge $[\mathbf{x}_i, \mathbf{x}_j]$ at time $t = t_n$. Equation (3.1) can then be expressed as

$$\rho_{\mathcal{V}}^{n+1} = (A_n + \delta t L_n)^{-1}(A_n \rho_{\mathcal{V}}^n). \tag{3.9}$$

To further explain the above discretization, note that the Laplace–Beltrami operator depends on the vertex positions on the sphere in the discrete case. As the vertex positions will be changed throughout the density-equalizing iterative process, the matrices A_n and L_n are also updated based on the deformed triangle mesh to reduce the numerical error.

After solving (3.9), we obtain the velocity field using (3.2). A discretization of the gradient $\nabla\rho$ for every triangular face $T = [\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k] \in \mathcal{F}$ at time t_n is given as follows (see [8] for details):

$$\nabla\rho_{\mathcal{F}}^n(T) = \frac{\mathbf{n}_{\mathcal{F}}^n(T) \times (\rho_{\mathcal{V}}^n(\mathbf{x}_i)e_{jk} + \rho_{\mathcal{V}}^n(\mathbf{x}_j)e_{ki} + \rho_{\mathcal{V}}^n(\mathbf{x}_k)e_{ij})}{2\mathrm{Area}(T)}, \tag{3.10}$$

where e_{ij}, e_{jk}, e_{ki} are the three directed edges $[\mathbf{x}_i(t_n), \mathbf{x}_j(t_n)]$, $[\mathbf{x}_j(t_n), \mathbf{x}_k(t_n)]$, $[\mathbf{x}_k(t_n), \mathbf{x}_i(t_n)]$, and $\mathbf{n}_{\mathcal{F}}^n(T)$ is the outward unit normal of the triangular face T. We can then easily convert $\nabla\rho_{\mathcal{F}}^n$ to $\nabla\rho_{\mathcal{V}}^n$ using the above-mentioned face-to-vertex conversion matrix M:

$$\nabla\rho_{\mathcal{V}}^n = M\nabla\rho_{\mathcal{F}}^n. \tag{3.11}$$

Now (3.2) becomes

$$\mathbf{v}_{\mathcal{V}}^n(\mathbf{x}_i) = -\frac{\nabla\rho_{\mathcal{V}}^n(\mathbf{x}_i)}{\rho_{\mathcal{V}}^n(\mathbf{x}_i)}. \tag{3.12}$$

As mentioned above, the velocity field $\mathbf{v}_{\mathcal{V}}^n$ may not lie exactly on the tangent space of the sphere due to discretization errors. To ensure that the vertices remain on the sphere under the diffusion process, we perform a projection of $\mathbf{v}_{\mathcal{V}}^n$ as follows:

$$\widetilde{\mathbf{v}}_{\mathcal{V}}^n(\mathbf{x}_i) = \mathbf{v}_{\mathcal{V}}^n(\mathbf{x}_i) - (\mathbf{v}_{\mathcal{V}}^n(\mathbf{x}_i) \cdot \mathbf{n}_{\mathcal{V}}(\mathbf{x}_i))\mathbf{n}_{\mathcal{V}}(\mathbf{x}_i), \tag{3.13}$$

where $\mathbf{n}_{\mathcal{V}}$ is the outward unit normal at the vertices. We can then update the position of all vertices $\mathbf{x}_i$ based on (3.3):

$$\mathbf{x}_i(t_{n+1}) = \mathbf{x}_i(t_n) - \delta t\ \widetilde{\mathbf{v}}_{\mathcal{V}}^n(\mathbf{x}_i). \tag{3.14}$$

While the projection of the velocity effectively eliminates the normal component, there may still be small numerical errors that cause $\mathbf{x}_i$ to move out of the spherical surface. This can be easily corrected by dividing $\mathbf{x}_i$ by its Euclidean 2-norm $\|\mathbf{x}_i\|_2$. Finally, as the number of iterations n increases, the density is equalized over the entire sphere; i.e., all values of $\rho_{\mathcal{V}}^n$ converge to their average.

3.1.3. Enforcing the bijectivity of the mapping throughout the iterative process. As discussed in [27], the bijectivity of the traditional density-equalizing mapping process is not guaranteed. Specifically, mesh overlaps may occur in the mapping result if the vertex displacement is too large. Analogously, in the spherical density-equalizing mapping process proposed

in the above section, (3.14) may cause some undesirable mesh overlaps under extremely large density gradients.

To resolve this issue, we propose an overlap correction scheme as follows. The main idea behind this scheme, as implied by Theorem 2.1, is to utilize quasi-conformal theory to rectify the fold-overs caused by the deformation in (3.14). For simplicity, we denote the initial conformal sphere as $\mathbb{S}_0^2$ and the resulting deformed sphere at the nth iteration as $\mathbb{S}_n^2$. To correct the overlaps on $\mathbb{S}_n^2$, we first apply the stereographic projection to map both $\mathbb{S}_n^2$ and $\mathbb{S}_0^2$ onto the plane, resulting in two corresponding planar domains referred to as D_n and D_0, respectively.

Note that the stereographic projection requires choosing a specific part of the sphere as the north pole. While the projection is conformal in theory, the conformality distortion of the triangle elements under the projection may be affected by the choice of the pole. Here, to reduce the conformality distortion of the projection, we select the triangle with the most regular 1-ring neighborhood as the north pole. More specifically, we first define the face regularity $R_{\mathcal{F}}^i$ for each triangle face T_i as follows:

$$R_{\mathcal{F}}^i = \sum_{j=1}^{3} \left| \frac{e_j^i}{e_1^i + e_2^i + e_3^i} - \frac{1}{3} \right|, \tag{3.15}$$

where e_1^i, e_2^i, e_3^i represent the length of the three edges of T_i. Then the vertex regularity can be defined by

$$R_{\mathcal{V}} = H^{\mathcal{FV}} R_{\mathcal{F}}. \tag{3.16}$$

Here $R_{\mathcal{V}}$ is a $|\mathcal{V}| \times 1$ matrix with the jth entry $R_{\mathcal{V}}^j$ being the vertex regularity of the jth vertex, $R_{\mathcal{F}}$ is an $|\mathcal{F}| \times 1$ face regularity matrix, and $H^{\mathcal{FV}}$ is a $|\mathcal{V}| \times |\mathcal{F}|$ sparse matrix such that

$$H_{ij}^{\mathcal{FV}} = \begin{cases} \frac{1}{3} & \text{if } T_j \text{ contains the } i\text{th vertex,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.17}$$

Finally, we determine the 1-ring face regularity $\overline{R}^i$ for each triangle T_i by calculating the average vertex regularity of the vertices in its 1-ring neighborhood:

$$\overline{R}^i = \frac{1}{3} \sum_{\mathbf{x}_j \in T_i} R_{\mathcal{V}}^j. \tag{3.18}$$

The most regular 1-ring triangular face is then chosen as the one with the lowest value of $\overline{R}^i$. Intuitively, the most regular 1-ring triangular face is a triangle element that is close to an equilateral triangle, with its neighboring triangles also being close to equilateral. Note that in theory, the stereographic projection maps geodesics between points on the sphere to circular arcs on the plane. However, in the discrete case, both the connections between the vertices on the spherical triangle mesh and the connections between the projected vertices on the plane are represented by straight line segments, and hence there will be a notable discrepancy for the triangles near the north pole, which causes numerical errors in the subsequent computation on the plane. By choosing the most regular 1-ring triangular face above as the north pole

for the stereographic projection, we can reduce the above-mentioned discrepancy and hence improve the accuracy of our computation.

After applying the stereographic projection to $\mathbb{S}_0^2$ and $\mathbb{S}_n^2$ using the above-mentioned approach, we can compute the Beltrami coefficient μ_n of the mapping between D_0 and D_n. By Theorem 2.1, the mapping is bijective if and only if the supremum norm of its Beltrami coefficient is less than 1. Therefore, the presence of mesh fold-overs can be determined by evaluating the norm of the Beltrami coefficient. In order to correct the overlaps, we introduce a truncation function $\mathbb{I}$ for the following Beltrami coefficient:

$$\mathbb{I}(\mu_n) = \begin{cases} \mu_n & \text{if } |\mu_n| < 1, \\ (1-\delta)\frac{\mu_n}{|\mu_n|} & \text{if } |\mu_n| \geq 1, \end{cases} \tag{3.19}$$

where δ is a prescribed truncation parameter. In practice, we set $\delta = 0.1$. For now, we neglect the outermost faces and only apply the truncation function to the faces at the central region of D_0. More specifically, the outermost faces are the triangles most distant from the origin on the planar domain, which also correspond to the triangles near the north pole of the sphere. As the geometric distortion of them caused by the stereographic projection is large, we first keep them fixed and only focus on correcting the mesh overlaps at the inner region of the planar domain. Denote the truncated Beltrami coefficient by $\widetilde{\mu}_n$. We can then obtain a new deforming map $\widetilde{g}_n$ by applying the following LBS algorithm [24]:

$$\widetilde{g}_n = \mathbf{LBS}(\widetilde{\mu}_n) \tag{3.20}$$

with the outermost region fixed. The updated spherical parameterization can be obtained through the inverse stereographic projection. Note that in some rare cases with a large number of mesh overlaps, the above truncation procedure may give a highly discontinuous Beltrami coefficient and hence still lead to a nonbijective reconstructed map. In such cases, one may halve the step size in (3.14) and repeat the above correction procedure.

Now note that the faces near the north pole were neglected in the above procedure. To also address potential overlaps of those faces, we project the updated spherical parameterization onto the complex plane using the south-pole stereographic projection. Then we can repeat the above procedure to correct the mesh overlaps using the Beltrami coefficient and finally obtain a bijective spherical parameterization using the inverse south-pole stereographic projection.

To further justify the bijectivity of the mapping under the above north pole–south pole overlap correction scheme, note that after applying the truncation function $\mathbb{I}$ and the LBS algorithm at the north-pole step, we obtain the updated map $\widetilde{g}_n$ with the Beltrami coefficient $\widetilde{\mu}_n$. The Jacobian matrix of $\widetilde{g}_n$ is given by

$$J_{\widetilde{g}_n} = |\,(\widetilde{g}_n)_z\,|^2 \left(1 - |\widetilde{\mu}_n|^2\right). \tag{3.21}$$

Since the truncation function ensures that $\|\widetilde{\mu}_n\|_\infty < 1$, the Jacobian matrix is positive. Consequently, the updated deforming map $\widetilde{g}_n$ contains no local mesh fold-overs at the inner region of the planar domain. Since the inverse stereographic projection is a diffeomorphism, there are no local mesh fold-overs at the southern region of the sphere after we map the planar result back to the sphere.

Similarly, after applying the south-pole stereographic projection and repeating the above process to truncate the Beltrami coefficient for the new inner region, the Jacobian of the resulting planar mapping is positive. Therefore, there are no mesh fold-overs at the inner region. Finally, by applying the inverse south-pole stereographic projection, the planar domain is mapped to the sphere again, and we can ensure that the northern region of the sphere, which corresponds to the above-mentioned inner region on the plane, does not contain mesh fold-overs. In other words, we conclude that the entire spherical mapping is folding-free.

By including this north pole–south pole overlap correction scheme at each iteration of the spherical density-equalizing mapping process, we can ensure that the bijectivity is preserved throughout the iterative process.

3.1.4. Recoupling the deformation and density throughout the iterative process. Recall that the proposed spherical density-equalizing mapping method uses density diffusion to produce shape deformations on the sphere. In particular, the density values and density gradients are converted between vertices and faces as described in (3.4) and (3.11) at each iteration, and the vertex positions are updated accordingly. However, in the discrete case, such a conversion may result in a smoothing effect, and hence numerical inaccuracies in the density may accumulate throughout the iterative process. Also, the vertex positions may be changed under the overlap correction scheme. Consequently, the shape deformation at each iteration may not perfectly correspond to the actual density flow. To address this issue, we propose an additional step that recouples the deformation and the density at the end of each iteration.

Specifically, at the end of the nth iteration, instead of directly passing the updated density obtained from (3.9) to the next iteration, we redefine the density on each triangular face $T = [\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k] \in \mathcal{F}$ using the updated map $\mathbf{x}(t_{n+1})$ as follows:

$$\rho_{\mathcal{F}}^{n+1}(T) = \frac{\text{Population}}{\text{Area}\left([\mathbf{x}_i(t_{n+1}), \mathbf{x}_j(t_{n+1}), \mathbf{x}_k(t_{n+1})]\right)}. \tag{3.22}$$

After getting the new density $\rho_{\mathcal{F}}^{n+1}$, we can update $n = n+1$ and proceed to (3.4) for the next iteration. In other words, instead of solving one single density diffusion equation over time and obtaining the shape deformation at every iteration based on it, we solve a new density diffusion equation at every iteration with both the density and the Laplace–Beltrami operator obtained based on the current mapping result and use the computed density gradient for one iteration only. With this additional recoupling step, the accuracy of the final mapping result can be improved.

3.1.5. Summary. Putting together the initial spherical conformal parameterization (section 3.1.1) and the iterative scheme for density equalization on the sphere (section 3.1.2) with the overlap correction scheme (section 3.1.3) and the recoupling scheme (section 3.1.4), we have the proposed SDEM algorithm for computing spherical density-equalizing maps. Following [8], we define the stopping criterion for the iterative scheme as $\frac{\text{sd}(\rho_{\mathcal{V}}^n)}{\text{mean}(\rho_{\mathcal{V}}^n)} < \epsilon$, where ϵ is a stopping parameter. The algorithm is summarized in Algorithm 3.1. In practice, the time step size is set to be $\delta t = 0.1$, the stopping parameter is set to be $\epsilon = 10^{-3}$, and the maximum number of iterations is $n_{\max} = 200$.

Algorithm 3.1 Bijective spherical density-equalizing map (SDEM).

Input: A genus-0 closed surface $\mathcal{M}$, a prescribed population, an initial spherical conformal parameterization $f_0 : \mathcal{M} \to \mathbb{S}^2$, the stopping parameter ϵ, and the maximum number of iterations allowed $n_{\max}$.

Output: A spherical density-equalizing map $f : \mathcal{M} \to \mathbb{S}^2$.

Compute the initial density $\rho_{\mathcal{F}}^0$ on $f_0(\mathcal{M})$ based on the prescribed population;
Set $n = 0$;
Let $\mathbf{x}_i(0) = f_0(v_i)$ for every vertex $v_i \in \mathcal{V}$;
repeat
  Compute A_n and L_n using (3.7) and (3.8);
  Obtain $\rho_{\mathcal{V}}^{n+1}$ by solving the diffusion equation (3.9);
  Compute the velocity field $\mathbf{v}_{\mathcal{V}}^{n+1}$ using (3.12);
  Compute the projected velocity field $\widetilde{\mathbf{v}}_{\mathcal{V}}^{n+1}$ using (3.13);
  Update the position of all $\mathbf{x}_i(t_{n+1})$ using (3.14);
  Apply the overlap correction scheme in section 3.1.3 to further update $\mathbf{x}_i(t_{n+1})$;
  Apply the recoupling scheme in section 3.1.4 to update $\rho_{\mathcal{F}}^{n+1}$ using (3.22);
  Update $n = n + 1$;
until $\frac{\mathrm{sd}(\rho_{\mathcal{V}}^n)}{\mathrm{mean}(\rho_{\mathcal{V}}^n)} < \epsilon$ or $n \geq n_{\max}$
The resulting spherical density-equalizing map is given by $f(v_i) = \mathbf{x}_i(t_n)$ for all i;

3.2. Bijective landmark-aligned spherical density-equalizing map (LSDEM).

In [27], it was shown that the planar density-equalizing mapping process can be reformulated as an energy minimization problem. Analogously, as the goal of our spherical density-equalizing mapping method is to transform the density gradient into shape deformations, the following energy is minimized throughout the mapping process:

$$E_{\mathrm{SDEM}} = \int \|\nabla \rho\|^2 . \tag{3.23}$$

More specifically, as our SDEM algorithm involves vertex position update, overlap correction, and the recoupling scheme at every iteration, each step can be considered as a change based on the descent direction of E_{SDEM} defined using the current vertex positions and the current density.

Moreover, it is worth noting that several prior works on spherical parameterization were also based on energy minimization. For instance, in [18], an iterative algorithm for computing spherical harmonic maps was developed by minimizing the following harmonic energy:

$$E_{\mathrm{Harmonic}} = \int \|\nabla f\|^2 . \tag{3.24}$$

As harmonic maps between genus-0 closed surfaces are conformal, minimizing E_{Harmonic} gives a spherical conformal parameterization. Another iterative algorithm was proposed by Lui et al. [25] to achieve optimized landmark-aligned spherical parameterizations. More specifically, an additional landmark mismatch energy was incorporated into their algorithm:

$$E_{\text{Landmark}} = \int \|f(p_i) - q_i\|^2, \tag{3.25}$$

where $\{p_i\}_{i=1}^k \leftrightarrow \{q_i\}_{i=1}^k$ are two sets of prescribed corresponding landmarks to be matched, with k being the total number of landmarks.

Motivated by the above approaches, here we aim to compute a bijective landmark-aligned spherical density-equalizing map (abbreviated as LSDEM) balancing density equalization, conformality, and landmark-matching conditions. This is achieved by minimizing the following proposed combined energy:

$$\begin{aligned} E &= \alpha\, E_{\text{SDEM}} + \beta\, E_{\text{Harmonic}} + \gamma\, E_{\text{Landmark}} \\ &= \alpha \int \|\nabla \rho\|^2 + \beta \int \|\nabla f\|^2 + \gamma \int \|f(p_i) - q_i\|^2, \end{aligned} \tag{3.26}$$

where α, β, γ are three nonnegative weighting parameters. Here, for the landmark-matching term $\gamma \int \|f(p_i) - q_i\|^2$, note that since all geodesic curves between points on the sphere $\mathbb{S}^2$ move along the great circle, the geodesic distance between two points decreases with their Euclidean distance. Therefore, we can simply use the standard Euclidean metric for the landmark-matching term to simplify our computation.

3.2.1. Descent direction for minimizing the proposed combined energy. To compute the landmark-aligned spherical density-equalizing map, we need to find a suitable descent direction to minimize the proposed combined energy in (3.26). In the following, we denote $E_1 = \alpha \int \|\nabla \rho\|^2$, $E_2 = \beta \int \|\nabla f\|^2$, $E_3 = \gamma \int \|f(p_i) - q_i\|^2$ for simplicity and determine the descent direction of them one by one.

We first consider the descent direction of $E_1 = \alpha \int \|\nabla \rho\|^2$. As explained in section 3.1, the velocity field induced by the density is given by $\mathbf{v} = -\frac{\nabla \rho}{\rho}$. In order to maintain the deformed shape as a sphere, we remove the normal component $\mathbf{v}^\perp = (\mathbf{v} \cdot \mathbf{n})\mathbf{n}$ of the velocity $\mathbf{v}$. The descent direction can then be defined using the following tangential velocity:

$$dE_1 = \alpha(\mathbf{v} - \mathbf{v}^\perp). \tag{3.27}$$

We remark that in this minimization problem for E_1, we do not employ the usual variational method. This is because the descent direction obtained through the variational method can potentially lead to a local minimum. By contrast, the velocity field we use is based on Fick's law of diffusion and can achieve the density-equalizing result, which corresponds to the global minimum of E_1.

For the second term $E_2 = \beta \int \|\nabla f\|^2$, the Euler–Lagrange equation can be derived as follows:

$$\begin{aligned} \left.\frac{d}{dt}\right|_{t=0} E_2(f + tg) &= \beta \int \left.\frac{d}{dt}\right|_{t=0} \|\nabla(f + tg)\|^2 \\ &= 2\beta \int \nabla f \nabla g \\ &= -2\beta \int \Delta f g. \end{aligned} \tag{3.28}$$

Note that the derivative in the above formula is negative when $g = \beta\Delta f$. To ensure that the deformed map remains a sphere, we again remove the normal component $\Delta^{\perp} f = (\Delta f \cdot \mathbf{n})\mathbf{n}$ of Δf and only use its tangential component $\Delta \bar{f} = \Delta f - \Delta^{\perp} f$. Now E_2 achieves its minimum if and only if the tangential component $\Delta \bar{f}$ vanishes, i.e., $\Delta f = \Delta^{\perp} f$. Therefore, we define

$$dE_2 = \beta \Delta \bar{f}. \tag{3.29}$$

For the landmark mismatch energy $E_3 = \gamma \int \|f(p_i) - q_i\|^2$, we compute its Euler–Lagrange equation:

$$\begin{aligned} \left.\frac{d}{dt}\right|_{t=0} E_3(f + tg) &= \gamma \int \left.\frac{d}{dt}\right|_{t=0} \|f(p_i) + tg(p_i) - q_i\|^2 \\ &= 2\gamma \int (f(p_i) - q_i) g(p_i). \end{aligned} \tag{3.30}$$

Similar to the above, we remove the normal component of $\left.\frac{d}{dt}\right|_{t=0} E_3(f + tg)$. The descent direction of E_3 can be expressed as follows:

$$dE_3(p) = \begin{cases} -\gamma(f(p_i) - q_i) & \text{if } p = p_i,\ i = 1, 2, \ldots, k, \\ 0 & \text{otherwise.} \end{cases} \tag{3.31}$$

Combining the above formulas, the descent direction for E is

$$dE = dE_1 + dE_2 + dE_3. \tag{3.32}$$

Hence, we can minimize the combined energy (3.26) by the following iterative scheme:

$$f_{n+1} = f_n + \delta t\ dE^n, \tag{3.33}$$

where δt is the time step size, f_n is the map at the nth iteration, and dE^n is the descent direction for E at the nth iteration.

3.2.2. Optimal rotation for further reducing the energy. While the iterative scheme described above can effectively reduce the proposed combined energy at every step, it is important to note that in practice, a large landmark mismatch can potentially result in mesh fold-overs during the deformation process. Specifically, a large landmark mismatch gives a large dE_3, and hence a large dE, which may subsequently lead to undesirable overlaps or intersections in the mesh and require extra effort to correct. Here we introduce an extra step of optimal rotation to further reduce the landmark mismatch energy throughout the iterative process, thereby enhancing the overall quality of the mapping.

First, we establish the following result.

Theorem 3.1. *The energies $E_{Harmonic}$ and E_{SDEM} are rotation invariant.*

Proof. The result follows from the fact that rotation is a metric-preserving rigid motion on the sphere. ∎

By Theorem 3.1, the first two terms in the proposed combined energy E in (3.26) are invariant under rotations. This allows us to consider finding an optimal rotation of the sphere

to reduce the landmark mismatch energy E_3 without affecting the two other energy terms in the combined energy E.

More specifically, let $\{f_n(p_i)\}_{i=1}^k$ be the current positions of the landmark vertices at the nth iteration of the iterative scheme, and let $\{q_i\}_{i=1}^k$ be their target positions on the sphere. We consider three rotation matrices R_x, R_y, R_z about the x-axis, y-axis, and z-axis, respectively:

$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix}, \quad R_y(\psi) = \begin{pmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{pmatrix}, \tag{3.34}$$
$$R_z(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where ϕ, ψ, θ are the rotation angles. We can then search for the optimal rotation angles that minimize the following landmark mismatch error:

$$L(\phi, \psi, \theta) = \sum_{i=1}^{k} \|R_x(\phi)R_y(\psi)R_z(\theta)f_n(p_i) - q_i\|^2. \tag{3.35}$$

Specifically, it is easy to see that

$$\frac{\partial L}{\partial \phi} = 2\sum_{i=1}^{k} \left\langle R_x R_y R_z f_n(p_i) - q_i, \frac{\partial R_x}{\partial \phi} R_y R_z f_n(p_i) \right\rangle, \tag{3.36}$$

$$\frac{\partial L}{\partial \psi} = 2\sum_{i=1}^{k} \left\langle R_x R_y R_z f_n(p_i) - q_i, R_x \frac{\partial R_y}{\partial \psi} R_z f_n(p_i) \right\rangle, \tag{3.37}$$

$$\frac{\partial L}{\partial \phi} = 2\sum_{i=1}^{k} \left\langle R_x R_y R_z f_n(p_i) - q_i, R_x R_y \frac{\partial R_z}{\partial \theta} f_n(p_i) \right\rangle, \tag{3.38}$$

and so one can easily obtain the optimal rotation angles $(\phi^*, \psi^*, \theta^*)$ using gradient descent. By applying the three optimal rotations $R_x(\phi^*)$, $R_y(\psi^*)$, $R_z(\theta^*)$ on the mapping f_n, the landmark mismatch energy E_{landmark} of the updated spherical mapping $\widetilde{f_n}$ will be less than or equal to that of f_n (see Figure 5 for an illustration). Moreover, since the two energies E_{Harmonic} and E_{SDEM} are unchanged under the rotation as proved in Theorem 3.1, it follows that the combined energy E of $\widetilde{f_n}$ is also less than or equal to that of f_n. In other words, this extra optimal rotation step can help us further reduce the landmark mismatch before performing the descent step. We can then replace f_n in section 3.2.1 with $\widetilde{f_n}$ and proceed with the descent step to obtain f_{n+1}.

3.2.3. Summary. Analogous to the SDEM algorithm, here in the proposed LSDEM algorithm, we start by computing an initial spherical conformal parameterization (section 3.1.1), from which we can get the initial density and the initial combined energy. Then we deform the initial spherical mapping using an iterative process consisting of both the optimal rotation step (section 3.2.2) and the descent step (section 3.2.1) for minimizing the combined energy.

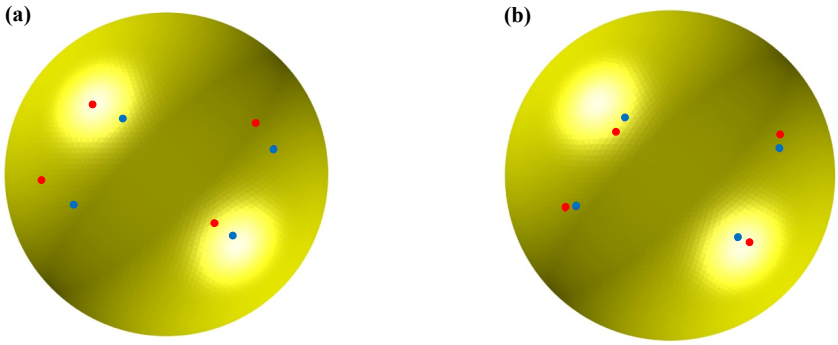

Figure 5. An illustration of the optimal rotation step in the proposed LSDEM algorithm. (a) *A spherical mapping with the landmarks (red) and their target positions (blue).* (b) *The optimal rotation result, in which the landmark mismatch is reduced.*

Algorithm 3.2 Landmark-aligned bijective spherical density-equalizing map (LSDEM).

Input: A genus-0 closed surface $\mathcal{M}$, a prescribed population, an initial spherical conformal parameterization $f_0 : \mathcal{M} \to \mathbb{S}^2$, the landmarks $\{p_1, \ldots, p_k\} \subset \mathcal{M}$ and target positions $\{q_1, \ldots, q_k\} \subset \mathbb{S}^2$, the stopping parameter ϵ, and the maximum number of iterations allowed $n_{\max}$.

Output: A landmark-aligned spherical density-equalizing map $f : \mathcal{M} \to \mathbb{S}^2$.

Compute the initial density $\rho_{\mathcal{F}}^0$ on $f_0(\mathcal{M})$ based on the prescribed population;
Set $n = 0$;
repeat
  Compute an optimal rotation of f_n as described in section 3.2.2 and obtain $\widetilde{f_n}$;
  Compute the descent direction dE^n by (3.32) and update $f_{n+1} = \widetilde{f_n} + \delta t \; dE^n$;
  Apply the overlap correction scheme in section 3.1.3 to further update f_{n+1};
  Apply the recoupling scheme in section 3.1.4 to update $\rho_{\mathcal{F}}^{n+1}$ using (3.22);
  Update $n = n + 1$;
until $\|f_n - f_{n-1}\| < \epsilon$ or $n \geq n_{\max}$
The resulting landmark-aligned spherical density-equalizing map is given by $f = f_n$;

The previously mentioned overlap correction scheme (section 3.1.3) and the recoupling scheme (section 3.1.4) are also included in the iterative process. We stop the iterations when the mapping result stabilizes: $\|f_n - f_{n-1}\| < \epsilon$, where ϵ is the stopping parameter. Ultimately, $f = f_n$ is the desired landmark-aligned bijective spherical density-equalizing map.

The proposed LSDEM method is summarized in Algorithm 3.2. In practice, the time step size is set to be $\delta t = 0.01$, the stopping parameter is set to be $\epsilon = 10^{-3}$, and the maximum number of iterations is $n_{\max} = 200$.

We remark that by changing the values of the parameters α, β, γ in the combined energy E, we can achieve different desired mapping effects in the LSDEM result. Specifically, by using a larger value of α, we can achieve a more density-equalizing result. By increasing the value of β, we can reduce the angle distortion in the mapping result. Last, by increasing the value of γ, we can reduce the landmark mismatch in the mapping result.

4. Experiments. In this section, we present experimental results to demonstrate the effectiveness of our proposed SDEM and LSDEM algorithms. The algorithms are implemented using MATLAB R2021a on the Windows platform. All experiments are conducted on a computer with an Intel(R) Core(TM) i9-12900 2.40 GHz processor and 32 GB memory. The surface meshes are from online mesh repositories [19]. All surfaces are discretized in the form of triangular meshes.

4.1. Spherical density-equalizing map. We start by considering some examples of mapping a spherical surface using the proposed SDEM method. In the example shown in Figure 6(a), we define different populations at different regions on the sphere to give a continuous initial density, with the maximum density and minimum density different by 10 times. We then apply the proposed method and obtain the spherical density-equalizing mapping result as shown in the figure, in which the high-density region is enlarged significantly and the low-density region is shrunk. By considering the initial and final density histograms, we can see that the density is effectively equalized using our method.

We then consider a more complicated spherical example involving multiple regions with discontinuous densities (Figure 6(b)). In particular, we define different populations on the sphere such that the initial density of all the triangular regions is three times the density of all the pentagonal regions. It can be observed that all the triangular regions expand and all the pentagonal regions shrink under the spherical density-equalizing map. We can also see from the initial and final density histograms that the density is effectively equalized using our method.

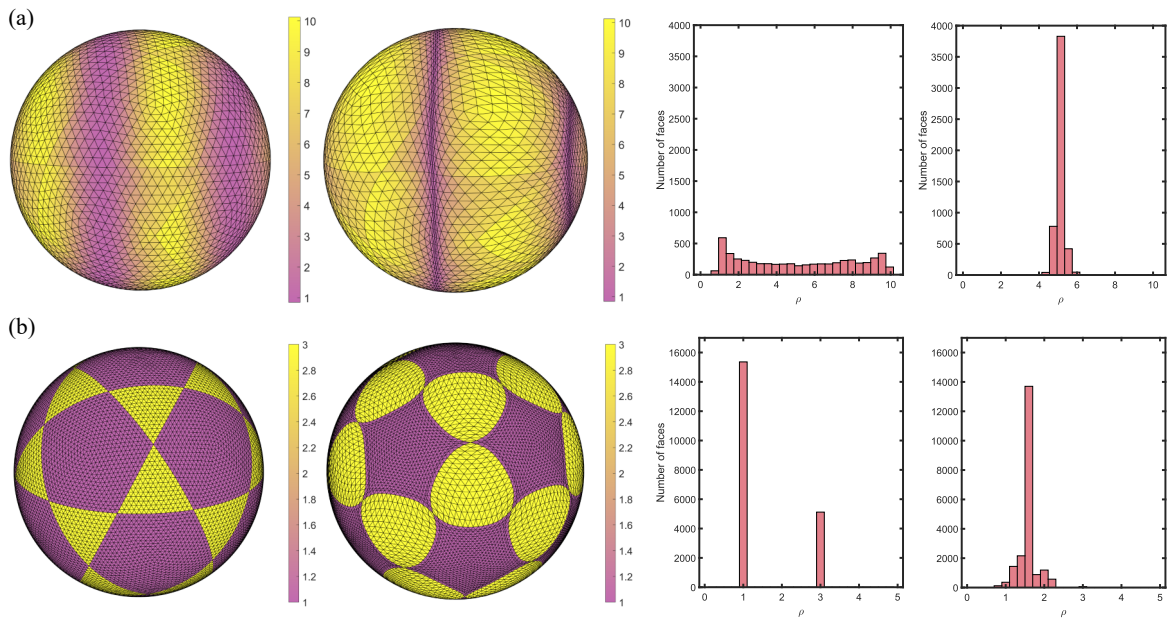

Figure 6. Spherical density-equalizing maps of spherical surfaces. *Each row shows one example.* (a) *An example with continuous input density.* (b) *An example with discontinuous input density. Left to right: the initial spherical surface color-coded with the initial density, the final SDEM result color-coded with the initial density, the histogram of the initial density, and the histogram of the final density.*

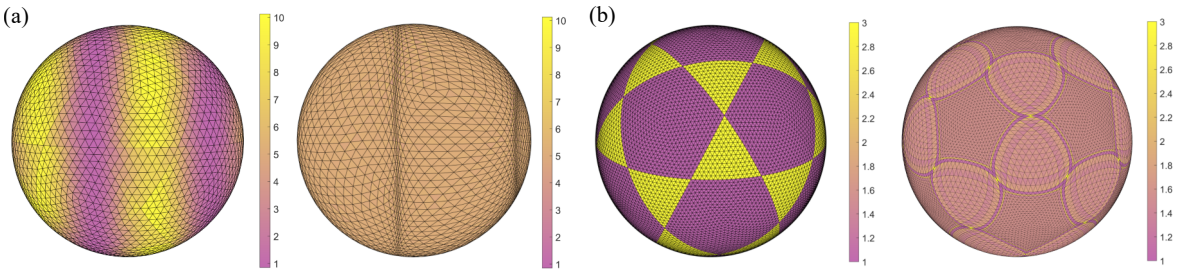

Figure 7. Alternative visualization of the spherical density-equalizing maps. (a) *The example in Figure 6(a) with continuous input density and the SDEM mapping result color-coded with the final density.* (b) *The example in Figure 6(b) with discontinuous input density and the SDEM mapping result color-coded with the final density.*

We remark that the mapping results in Figure 6 are color-coded with the initial density such that the deformations of different regions corresponding to different initial density values can be easily visualized. In Figure 7, we provide an alternative visualization of the mapping results color-coded with the final density, from which we can see that the final density is effectively equalized over the entire domain. In particular, for the continuous input density example in Figure 7(a), the final density becomes uniform in both the magnified and the shrunk regions. For the discontinuous input density example in Figure 7(b), the final density also becomes highly uniform everywhere except for the interface between different density regions, which can be explained by the fact that the diffusion equation assumes the density to be differentiable while the input density here is discontinuous.

Also, recall that we set the stopping parameter in Algorithm 3.1 as $\epsilon = 10^{-3}$ in practice. To further study the convergence behavior of our proposed SDEM method, in Figure 8 we show the change in the density error throughout the iterative process and plot the density histograms of the mapping right after reaching $\epsilon = 10^{-1}$, 10^{-2}, 10^{-3}, 10^{-4}, and 10^{-5}. It can be observed that the result stabilizes rapidly, and further running the iterations will only improve the result marginally. Therefore, we set the stopping parameter as $\epsilon = 10^{-3}$ to balance between accuracy and efficiency in practice.

Next, we consider computing spherical density-equalizing maps for general genus-0 surfaces. In Figure 9(a), we consider mapping the Duck model with the head part of it enlarged. As shown in the figure, we can apply our proposed SDEM method and compute a spherical density-equalizing map with the desired effect. The histograms show that the density is effectively equalized. Similarly, as shown in Figure 9(b), we can effectively magnify and shrink different parts in the mapping result of the David model. In particular, one of its eyes is enlarged significantly while the other one is shrunk.

For a more quantitative analysis of our proposed SDEM method, in Table 1 we record computational time, variance of the initial density, variance of the final density, and number of overlaps for mapping different surface models. It can be observed that under our SDEM algorithm, the variance in the density can be significantly reduced. Also, from the number of overlaps, we can see that the mappings are all bijective. As for the computational time, we can see that our algorithm is highly efficient and only takes a few seconds for all examples.

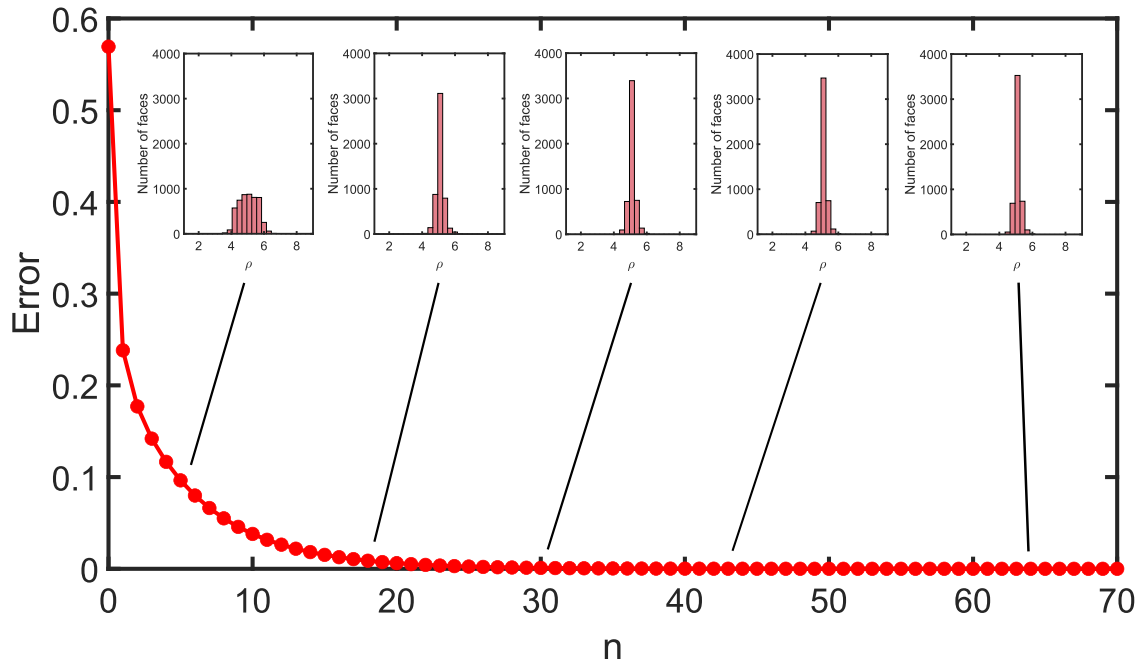

Figure 8. The change in the density error throughout the iterative process in the proposed SDEM method. *Here we run the proposed method on the example in Figure* 6(a) *and record the change in the density error. The insets show the density histograms of the mapping right after reaching* $\epsilon = 10^{-1}$, 10^{-2}, 10^{-3}, 10^{-4}, *and* 10^{-5}, *respectively.*

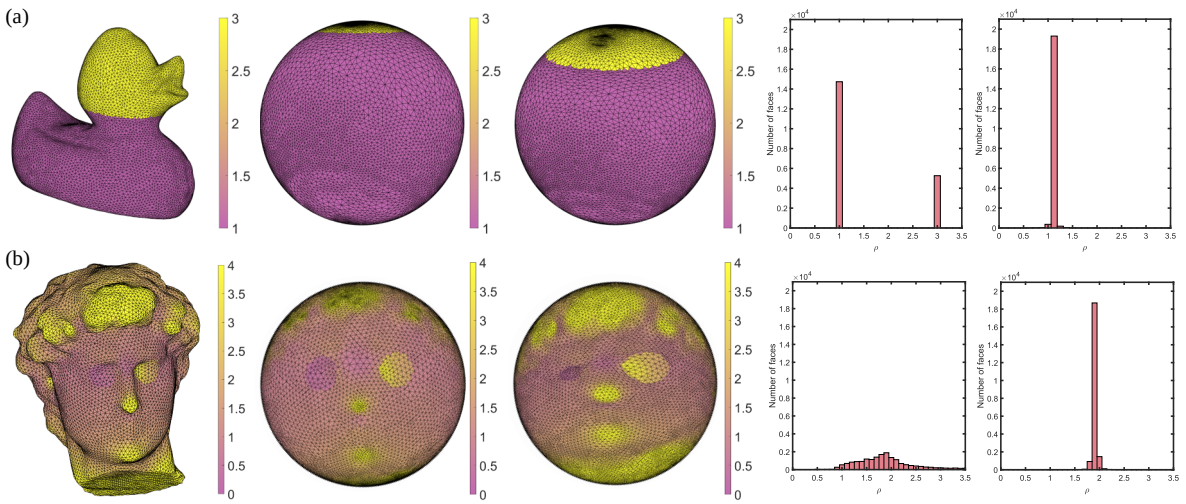

Figure 9. Spherical density-equalizing maps of genus-0 closed surfaces. *Each row shows one example.* (a) *The Duck model.* (b) *The David model. Left to right: the input surface mesh color-coded with the initial density, the spherical conformal parameterization color-coded with the initial density, the final SDEM result color-coded with the initial density, the histogram of the initial density, and the histogram of the final density.*

Table 1

The performance of our SDEM algorithm. *For each surface, we record the number of triangle elements, the computational time, the variance of the normalized initial density* $\widetilde{\rho}_1 = \frac{\rho_1}{Mean(\rho_1)}$, *the normalized final density* $\widetilde{\rho}_2 = \frac{\rho_2}{Mean(\rho_2)}$, *where* ρ_1 *is the initial vertex density and* ρ_2 *is the final vertex density, and the number of overlaps.*

Surface	# Faces	Time (s)	$\mathbf{Var}(\widetilde{\rho}_1)$	$\mathbf{Var}(\widetilde{\rho}_2)$	# Overlaps
Sphere 1 (Figure 6(a))	5120	0.4471	0.3236	< 0.0001	0
Sphere 2 (Figure 6(b))	20480	0.2443	0.2935	0.0015	0
Duck (Figure 9(a))	20000	1.4416	0.3294	< 0.0001	0
David (Figure 1, enlarge)	21338	1.4281	0.6099	< 0.0001	0
David (Figure 1, shrink)	21338	1.4295	0.6004	< 0.0001	0
David (Figure 9(b), mixed)	21338	1.4267	0.5949	< 0.0001	0

Note that by setting the population as the face area of the input mesh and applying the SDEM method, we can achieve a spherical area-preserving parameterization. To illustrate this, we consider several genus-0 closed surfaces as shown in Figure 10. For each surface, we first parameterize it onto the sphere using the spherical conformal parameterization. Then we apply the SDEM method to obtain an area-preserving parameterization. To assess the area-preserving quality of the spherical parameterizations, we can evaluate the area distortion by considering the logged area ratio for every triangular face T:

$$d_{\text{area}}(f)(T) = \log\left(\frac{\text{Area}(f(T))/\sum_{T'\in\mathcal{F}}\text{Area}(f(T'))}{\text{Area}(T)/\sum_{T'\in\mathcal{F}}\text{Area}(T')}\right), \tag{4.1}$$

where f is the spherical parameterization and $\mathcal{F}$ is the set of all triangular faces. Note that the two factors in the numerator and denominator are used for normalization so that a perfectly area-preserving parameterization would yield $d_{\text{area}} \equiv 0$. From the results in Figure 10, it can be observed that our SDEM method can handle different genus-0 closed surfaces with highly complex geometry very well. Specifically, the initial area distortion histograms in Figure 10 show that the area distortion is very large in the initial conformal parameterization for all examples. By contrast, the final area distortion histograms are all highly concentrated, which indicates that the final mappings are highly area-preserving. For a more quantitative comparison, Table 2 records the computational time, the initial and final area distortions, and the number of overlaps for different surfaces. In particular, it can be observed that our SDEM algorithm effectively reduces the area distortion by over 90% in all experiments while maintaining the bijectivity of the mappings. This shows that our algorithm is capable of computing the spherical area-preserving parameterization for a wide range of genus-0 closed surfaces.

Recall that we have introduced an overlap correction scheme in our proposed SDEM method to enforce the bijectivity of the mapping throughout the density-equalizing iterative process. It is natural to ask how important this additional step is to the final density-equalizing mapping results. In Figure 11, we compare the mapping results obtained using our SDEM method and SDEM without including the overlap correction scheme. It can be observed that there are multiple overlaps in the mapping results obtained by SDEM without the overlap correction scheme, especially at the regions corresponding to certain sharp features in the input

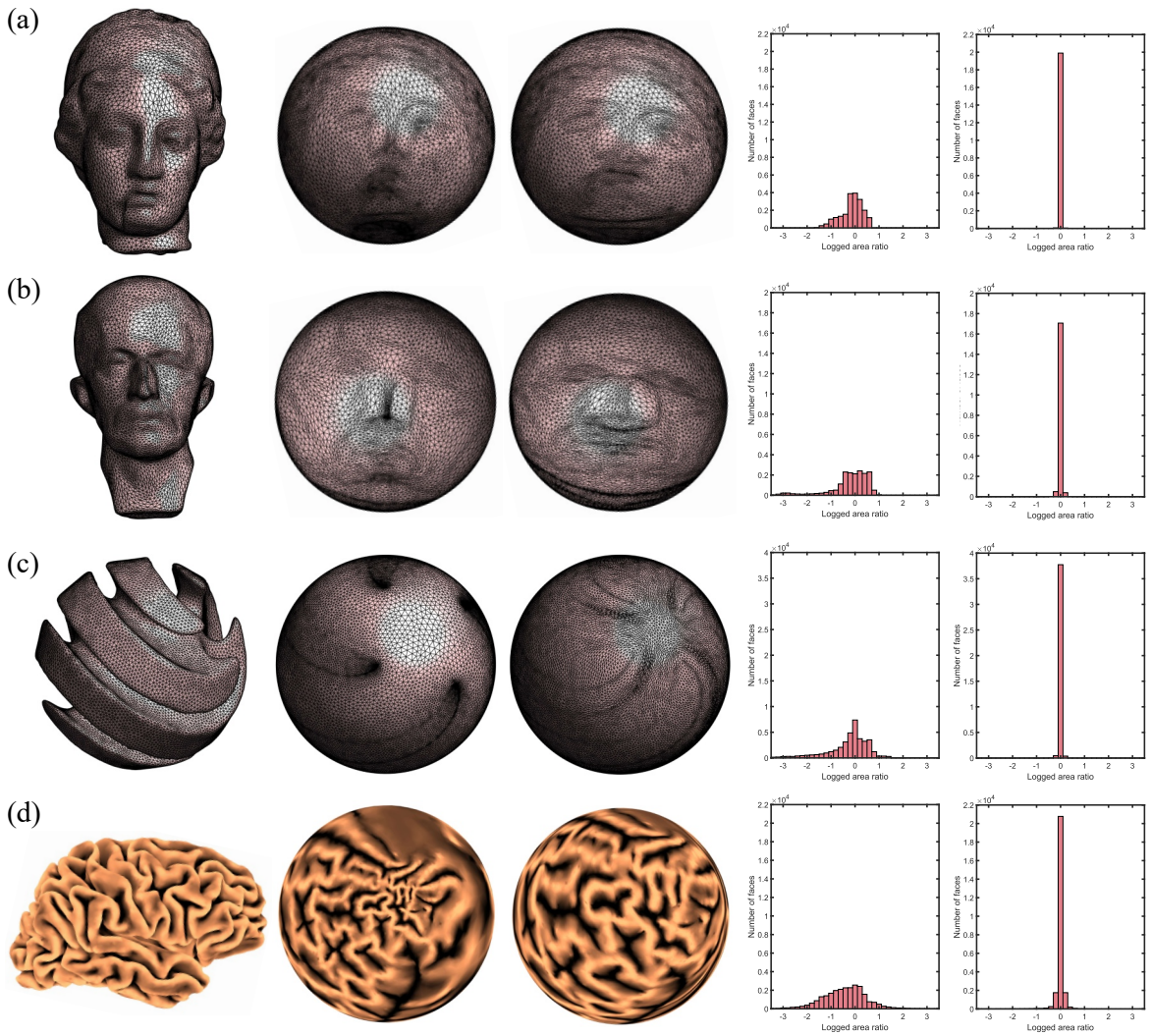

Figure 10. Spherical area-preserving parameterization of genus-0 closed surfaces obtained by the SDEM method. *Each row shows one example.* (a) *The Igea model.* (b) *The Max Planck model.* (c) *A twisted ball.* (d) *A brain cortical surface. Left to right: the input surface mesh, the initial spherical conformal parameterization, the final SDEM result, the histogram of the logged area ratio d_{area} of the initial spherical parameterization, and the histogram of the logged area ratio d_{area} of the final spherical parameterization. The surfaces in* (d) *are color-coded with the brain surface curvature for better visualization.*

surface. By contrast, the mapping results obtained using SDEM with the overlap correction scheme are bijective. This demonstrates the significance of the overlap correction scheme in our proposed method for ensuring the bijectivity of the spherical density-equalizing mapping results.

4.2. Landmark-aligned spherical density-equalizing map. After demonstrating the effectiveness of our proposed SDEM method, we consider our proposed LSDEM method. Figure 12(a)–(b) shows two examples of mapping a spherical surface with landmark constraints

Table 2

The performance of our SDEM algorithm for spherical area-preserving parameterization. *For each surface, we record the number of triangle elements, the computational time, the mean and standard deviations of the initial area distortion $|d_{area}(f_0)|$ and the final area distortion $|d_{area}(f)|$, and the number of overlaps.*

			$\lvert d_{\mathrm{area}}(f_0)\rvert$		$\lvert d_{\mathrm{area}}(f)\rvert$		
Surface	**# Faces**	**Time (s)**	**Mean**	**SD**	**Mean**	**SD**	**# Overlaps**
Chinese Lion (Figure 1)	10000	6.2777	1.6837	1.6941	0.1199	0.1786	0
Igea (Figure 10(a))	20000	1.1320	0.3668	0.3163	0.0146	0.0183	0
Max Planck (Figure 10(b))	18000	1.0756	0.5619	0.6252	0.0296	0.0411	0
Twisted Ball (Figure 10(c))	38620	1.5043	0.6240	0.7174	0.0210	0.0294	0
Brain (Figure 10(d))	25000	4.5765	0.7514	0.6222	0.0698	0.1414	0
Frog (Figure 15)	20000	2.9655	0.9056	1.3382	0.0460	0.0822	0

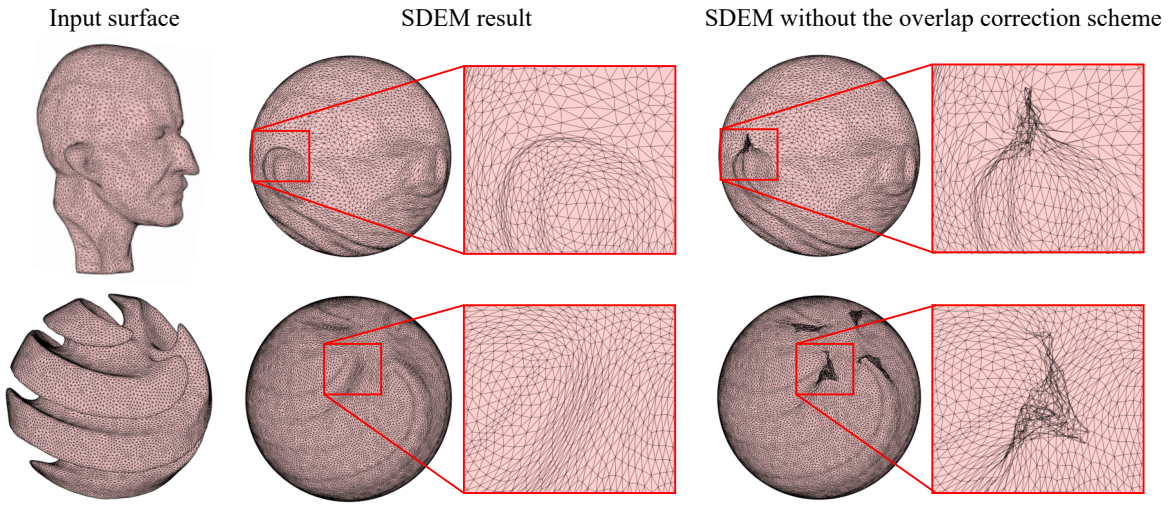

Figure 11. Comparison between the proposed SDEM method and SDEM without the overlap correction scheme. *Each row shows one set of examples. (Left) The input surface. (Middle) The spherical density-equalizing mapping result obtained by our proposed SDEM method. (Right) The mapping result obtained by running the SDEM method for the same number of iterations but with the overlap correction scheme excluded. The zoom-in highlights the difference between the results.*

using the proposed LSDEM method, and Figure 12(c)–(e) shows several examples of mapping general genus-0 closed surfaces with prescribed landmark constraints. It can be observed in all examples that by using the proposed LSDEM method, we can obtain spherical mapping results with different regions enlarged or shrunk suitably and with all landmarks being well aligned.

To analyze the performance of our LSDEM algorithm more quantitatively, Table 3 shows the computational time, the variance of the initial density and final density, the norm of the Beltrami coefficient, the initial and final landmark mismatch error, and the number of overlaps for mapping different surfaces. Note that because of the additional landmark constraints in the LSDEM problem formulation, it is expected that the density-equalizing effect will not be

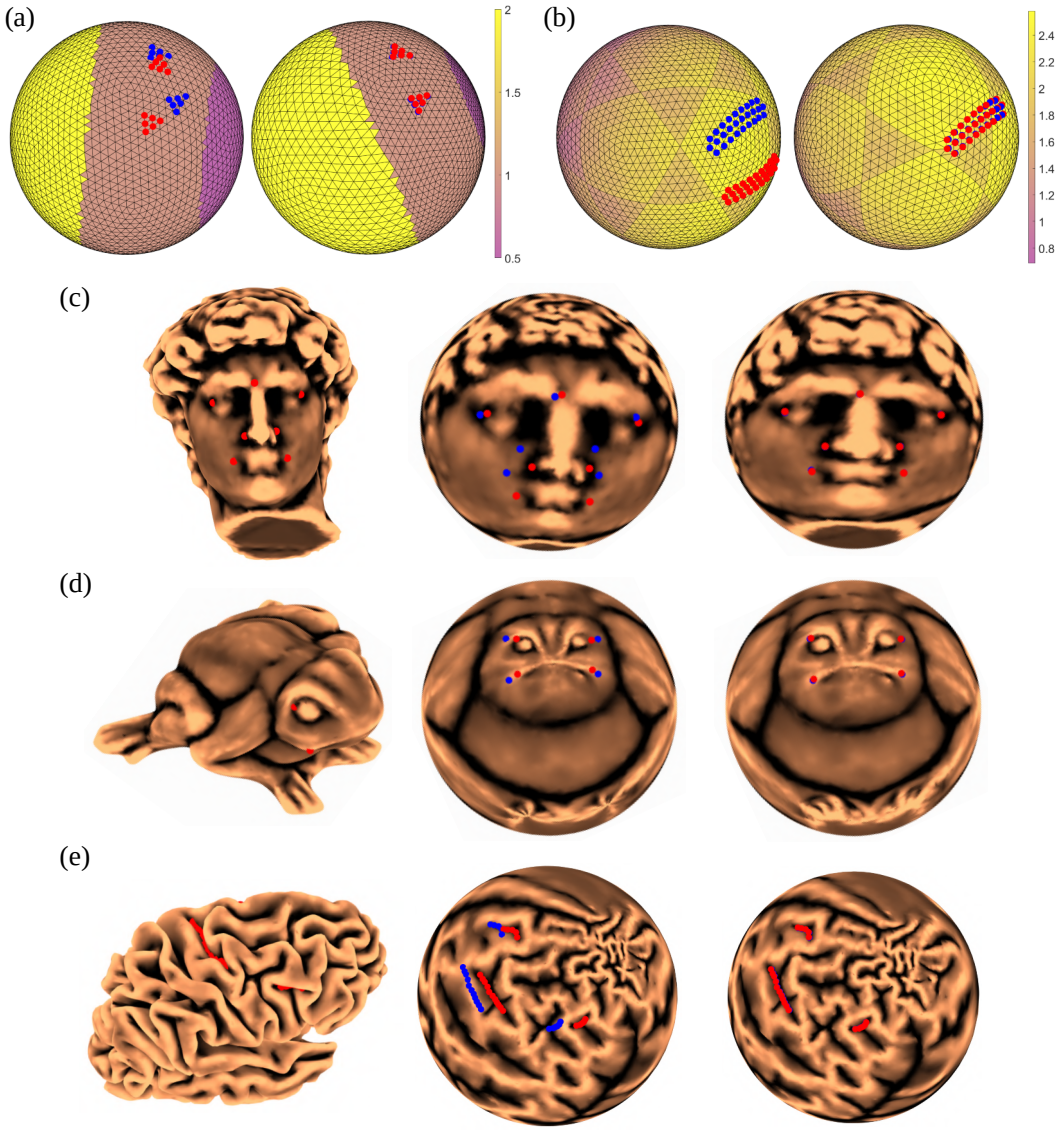

Figure 12. Landmark-aligned spherical density-equalizing maps of genus-0 closed surfaces. (a)–(b) *Two examples of mapping a spherical surface with different prescribed populations and landmark constraints. In each example, the left panel shows the initial spherical surface color-coded with the initial density together with the labeled landmarks (red dots) and the prescribed target positions (blue dots), and the right panel shows the final LSDEM result.* (c)–(e) *Examples of mapping general genus-0 closed surfaces. Left to right: the input genus-0 surface color-coded with the surface mean curvature and the labeled landmarks (red dots), the initial spherical conformal parameterization together with the labeled landmarks (red dots) and the prescribed target positions (blue dots), and the final LSDEM result. In all examples, the parameters in the energy* (3.26) *are set as* $(\alpha, \beta, \gamma) = (1, 2, 5)$.

as good as the one achieved by SDEM. Nevertheless, we can see in all experimental results that the variance of the density is significantly reduced under the LSDEM algorithm, which indicates that the mappings are close to density-equalizing. We also see that the landmark

Table 3

The performance of our LSDEM algorithm. *For each surface, we record the number of triangle elements, the computational time, the variance of the normalized initial density* $\widetilde{\rho}_1 = \frac{\rho_1}{Mean(\rho_1)}$ *and the normalized final density* $\widetilde{\rho}_2 = \frac{\rho_2}{Mean(\rho_2)}$*, where* ρ_1 *is the initial vertex density and* ρ_2 *is the final vertex density, the mean of the norm of the Beltrami coefficient* $|\mu|$*, the* 2*-norm of the landmark error in the initial map* $\mathbf{LE}_1$*, the* 2*-norm of the landmark error in the final result* $\mathbf{LE}_2$*, and the number of overlaps. In all examples, the parameters in the energy* (3.26) *are set as* $(\alpha, \beta, \gamma) = (1, 2, 5)$*.*

Surface	# Faces	Time (s)	$\mathbf{Var}(\widetilde{\rho}_1)$	$\mathbf{Var}(\widetilde{\rho}_2)$	$\mathbf{Mean}(\lvert\mu\rvert)$	$\mathbf{LE}_1$	$\mathbf{LE}_2$	# Overlaps
Sphere 1 (Figure 12(a))	5120	1.6780	0.1857	0.0077	0.0788	2.7450	0.0897	0
Sphere 2 (Figure 12(b))	5120	1.2989	0.0873	0.0013	0.0479	15.9647	0.2501	0
David (Figure 12(c))	21338	13.1052	0.5962	0.0118	0.1470	0.9222	0.0405	0
Frog (Figure 12(d))	20000	10.8065	26.6130	0.6132	0.1411	0.2913	0.0459	0
Brain (Figure 12(e))	25000	16.5567	1.4136	0.6063	0.0876	3.9402	0.0188	0

Table 4

The performance of our LSDEM algorithm with different parameters. *Here* α, β, γ *are the parameters in the energy* (3.26)*. See the caption of Table* 3 *for the description of the other columns.*

α	β	γ	$\mathbf{Var}(\widetilde{\rho}_2)$	$\mathbf{Mean}(\lvert\mu\rvert)$	$\mathbf{LE}_2$	# Overlaps
1	1	1	0.0027	0.1687	0.2412	0
3			0.0009	0.1720	0.3006	0
5			0.0004	0.1697	0.2292	0
1	1	1	0.0027	0.1687	0.2412	0
	3		0.0159	0.1468	0.1670	0
	5		0.0573	0.1195	0.2576	0
1	1	1	0.0027	0.1687	0.2412	0
		3	0.0024	0.1690	0.1939	0
		5	0.0064	0.1527	0.0714	0

mismatch error is very small and the quasi-conformal distortion is low in all examples. Moreover, the number of overlaps is again 0 in all experiments, which indicates that our LSDEM results are all bijective.

It is natural to ask how the choices of the parameters α, β, γ in the energy (3.26) will affect the mapping results in the proposed LSDEM algorithm. Here we consider mapping the David model with different values of α, β, γ used and analyzing the mapping results in terms of the density-equalizing effect, quasi-conformal distortion, landmark mismatch error, and bijectivity. As shown in Table 4, if we increase the value of α while keeping β and γ fixed, we will be able to reduce the variance of the final density. In other words, we will achieve a better density-equalizing effect. By contrast, if we increase the value of β while keeping α and γ fixed, the quasi-conformal distortion will be reduced. If we increase the value of γ while keeping α and β unchanged, we will achieve a lower landmark mismatch error. It is

noteworthy that the mapping results contain no overlaps for all combinations of α, β, γ, which indicates that the bijectivity of the mappings is ensured under our LSDEM method.

5. Applications. In this section, we introduce the applications of our proposed SDEM and LSDEM methods for surface registration, surface remeshing, and spherical data visualization.

5.1. Surface registration. Suppose $\mathcal{M}$, $\mathcal{N}$ are two genus-0 closed surfaces with some corresponding features $\{p_i\}_{i=1}^k \leftrightarrow \{q_i\}_{i=1}^k$. Using the proposed LSDEM method, we can find two spherical parameterizations $f : \mathcal{M} \to \mathbb{S}^2$ and $g : \mathcal{N} \to \mathbb{S}^2$ with the landmarks well aligned (i.e., $f(p_i) \approx g(q_i)$ for all $i = 1, 2, \ldots, k$). Then we can use the inverse mapping to build a 1-1 correspondence between $\mathcal{M}$ and $\mathcal{N}$. More explicitly, the composition $g^{-1} \circ f : \mathcal{M} \to \mathcal{N}$ will be a landmark-aligned mapping between the two surfaces. Figure 13 shows an example of registering the Igea model and the Max Planck model, in which we use the eyes and noses as landmarks for computing the landmark-aligned spherical parameterizations. We can then obtain the registration mapping from Igea onto Max Planck, with the eyes and noses well matched in the registration result.

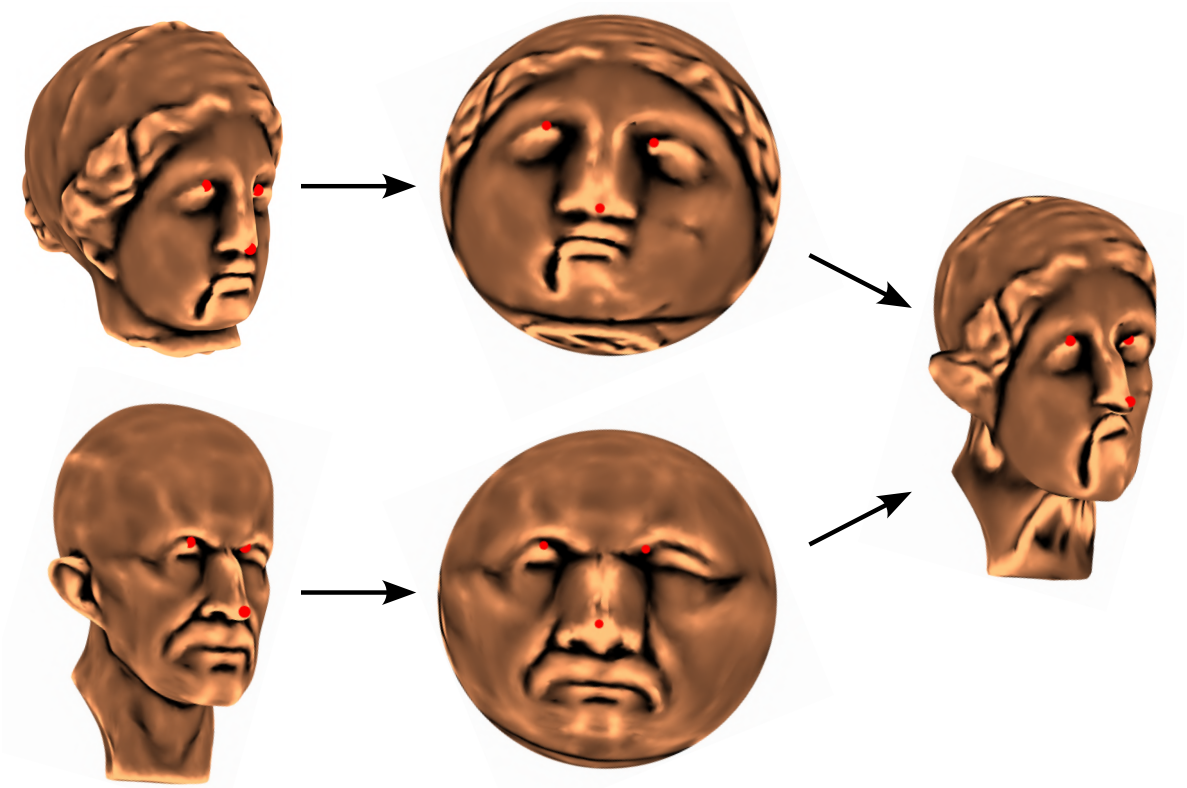

Figure 13. Surface registration via our proposed LSDEM method. *We first label some consistent features on both the Igea model and the Max Planck model as landmarks. Then we compute a landmark-aligned spherical density-equalizing map for each model, with the two sets of landmarks mapped to consistent locations on the unit sphere. We can then use the spherical parameterization to register the Igea model with the Max Planck surface, with the eyes and noses well aligned.*

5.2. Surface remeshing. Using the proposed SDEM method, we can perform surface remeshing for genus-0 closed surfaces easily. More specifically, given a genus-0 closed surface $\mathcal{M}$ to be remeshed, we can first compute a spherical density-equalizing map $f : \mathcal{M} \to \mathbb{S}^2$. Then we can generate a uniform mesh on the sphere using DistMesh [31]. Using the inverse mapping f^{-1}, we can then map the uniform mesh structure on the sphere back onto the given surface $\mathcal{M}$, which gives a remeshed surface.

In particular, by computing an area-preserving parameterization using SDEM, we can ensure that the mesh density will be largely uniform in the surface remeshing result. Figure 14 shows several examples of remeshing the Max Planck model with different numbers of vertices. It can be observed that the mesh quality of the remeshed surfaces is very high, with the triangle elements being uniform in size and distribution. In Figure 15, we further compare our remeshing result with the remeshing result obtained using conformal parameterization [6, 10]. It can be observed that the triangle elements on the remeshed surface obtained via

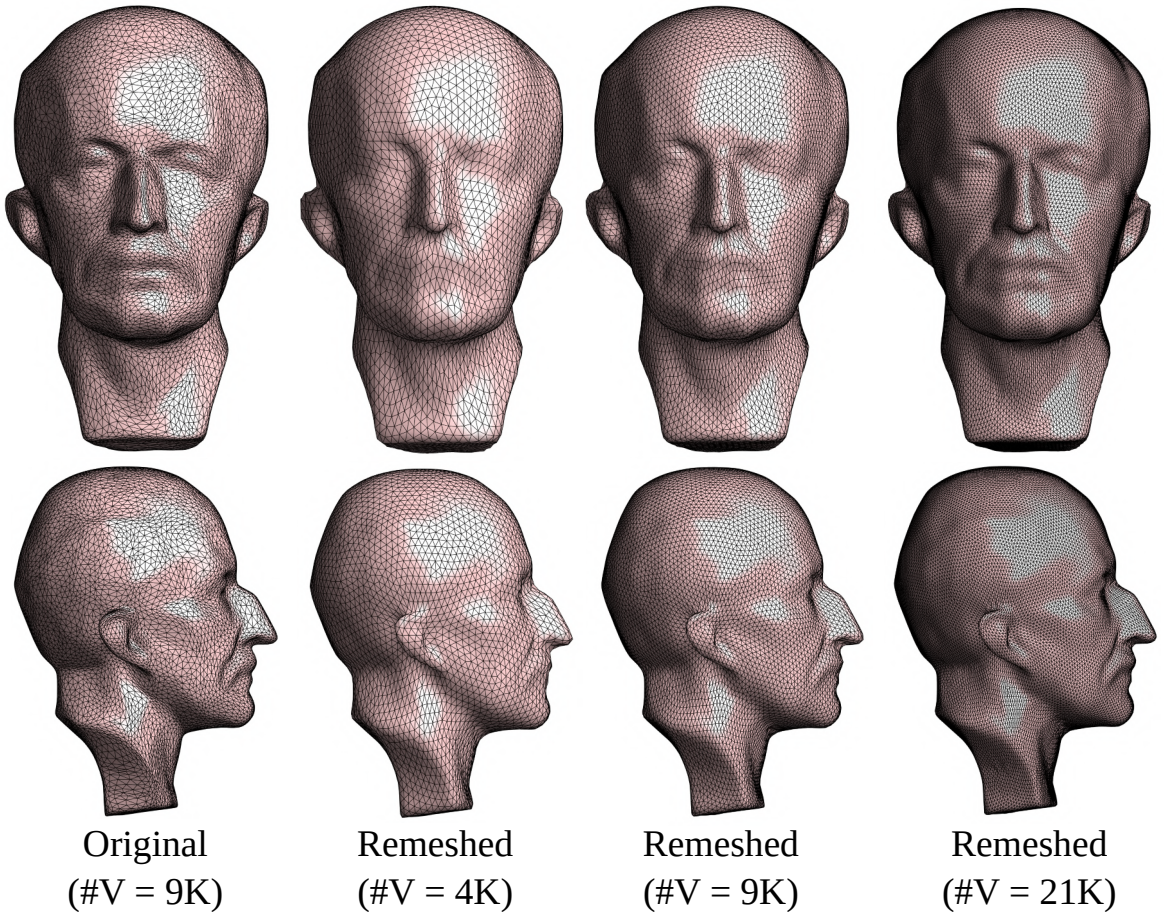

Figure 14. Surface remeshing via our proposed SDEM method. *The first column shows the front view and side view of the original Max Planck model. The second, third, and fourth columns show the front view and side view of three remeshed surfaces with different numbers of vertices obtained via SDEM.*

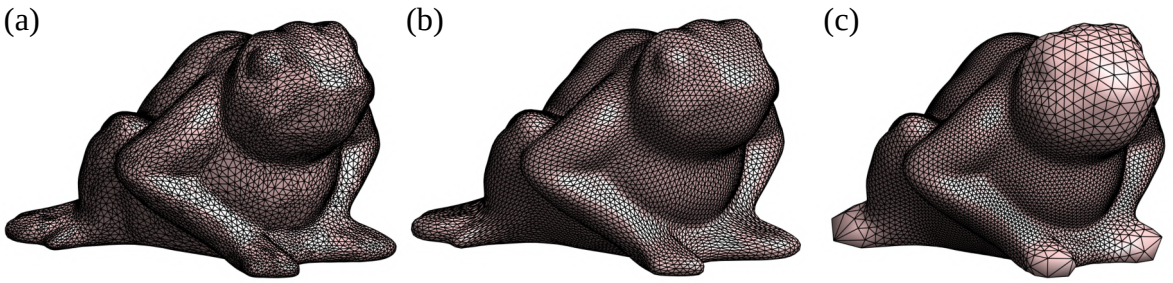

Figure 15. A comparison between our SDEM method and conformal parameterization for surface remeshing. (a) *The original Frog model.* (b) *A remeshed surface obtained via spherical area-preserving parameterization by SDEM.* (c) *A remeshed surface obtained via spherical conformal parameterization* [6, 10]. *The two remeshed surfaces contain the same number of vertices and faces.*

our method are much more uniform than those obtained via conformal parameterization, which can be explained by the fact that the conformal parameterizations preserve angles but may yield a large distortion in area. Moreover, the sharp features such as the head and the feet of the Frog model are much more well preserved under our approach. This shows that our method is more advantageous for surface remeshing.

One can also control the mesh density at a certain region by setting an appropriate population in the computation of SDEM. Specifically, if a higher mesh density is desired at a certain region, we can set a larger population and run the SDEM algorithm. Then the region will be enlarged in the resulting spherical parameterization. Consequently, if we generate a uniform mesh on the sphere using DistMesh and map the new mesh structure back onto the given surface, the mesh density at the corresponding region on the given surface will be higher than the other regions. As an example, Figure 16(a) shows a remeshed surface of the Buddha model obtained by computing a spherical area-preserving parameterization using SDEM, in which we can see that the triangle elements are uniform in size and distribution. By contrast, Figure 16(b) shows another remeshed surface obtained by computing a spherical density-equalizing map with a larger population set at the head region in the SDEM computation. It can be observed that since the head region is enlarged in the resulting spherical parameterization, the mesh density at the head region of the new mesh structure induced by the inverse mapping will be naturally higher than that at other parts of the remeshed surface.

5.3. Spherical data visualization. Analogous to the traditional DEM method [15], the SDEM method can be applied for visualizing sociological data on a spherical domain (see Figure 17 for an illustration). Specifically, given a spherical earth model, we can set the initial density on the underlying spherical triangle mesh based on certain sociological data and compute a spherical density-equalizing map. Then we can use the density-equalizing mapping result to obtain a deformed earth model, which provides us with an intuitive way to visualize, understand, and analyze the data. Moreover, as the mappings produced by the SDEM method are bijective, no regions will be overlapping in the deformed models.

As an example, in Figure 18 we compute three spherical density-equalizing maps based on the world population in Years 1980, 2000, and 2020, respectively. More specifically, for

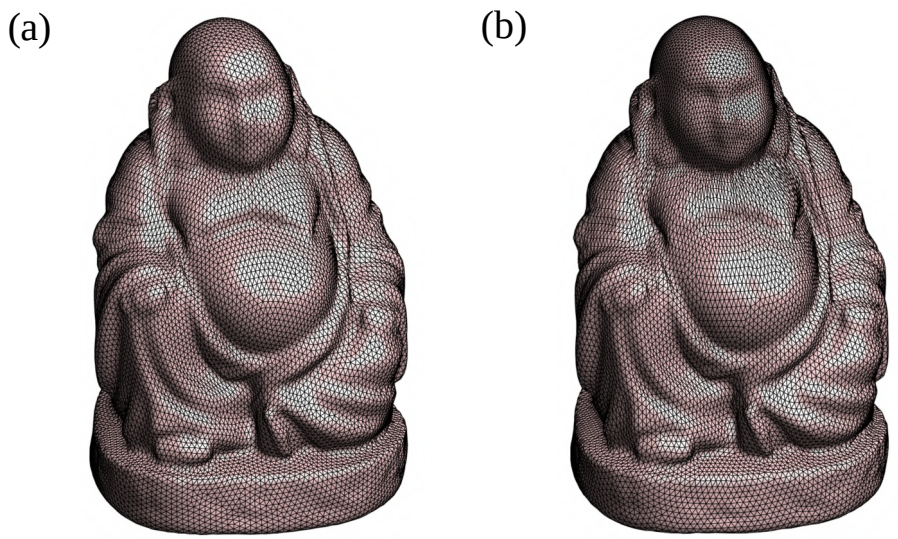

Figure 16. Controlling the mesh density in surface remeshing via SDEM. (a) *A remeshed surface of the Buddha model obtained via spherical area-preserving parameterization by SDEM.* (b) *A remeshed surface of the Buddha model obtained via spherical density-equalizing map with the head part enlarged by SDEM. The two remeshed surfaces contain the same number of vertices and faces.*

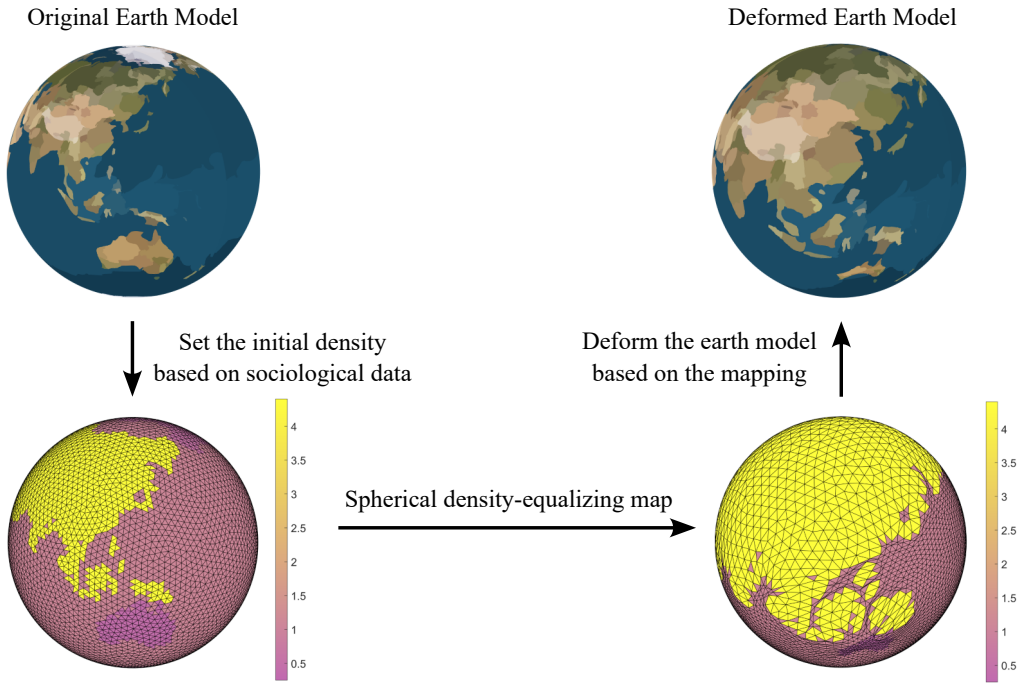

Figure 17. An illustration of the spherical data visualization process. *Given a spherical earth model (top left), we first set the initial density on the underlying spherical triangle mesh based on some sociological data (bottom left). We then compute a spherical density-equalizing map (bottom right) and deform the earth model based on the mapping result, thereby producing a spherical cartogram for data visualization (top right). Here both the original spherical triangle mesh and the mapping result are color-coded with the initial density.*

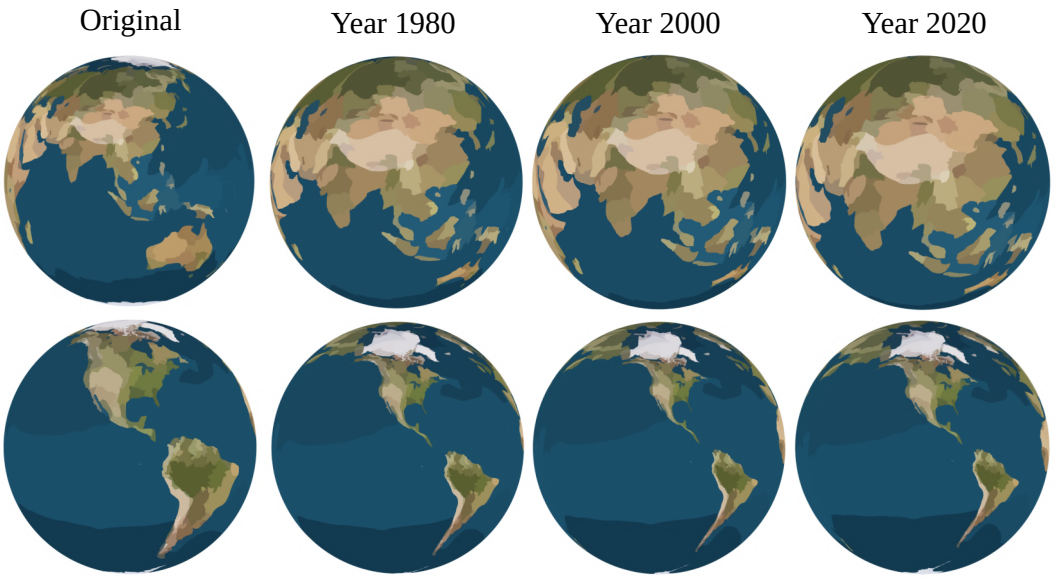

Figure 18. Visualizing the change in the world population via SDEM. *The first column shows the undeformed earth model. The second, third, and fourth columns show the deformed earth models produced by SDEM based on the world population in Years* 1980, 2000, *and* 2020, *respectively. Two different views are displayed for each earth model (top row and bottom row).*

each set of world population data in a specific year, we set the initial densities at different regions (Asia, Africa, North America, South America, etc.) of the spherical earth model to be proportional to the actual human population in those regions. Also, we set the density at the sea to be the average of the density at all other regions. Then, under the proposed SDEM algorithm, regions with a higher actual human population will expand and those with a lower actual human population will shrink. As the human population in Asia is higher than that in North America and South America, we can see that the Asia region is magnified in all three mapping results in Figure 18, while the North America and South America regions are shrunk. Moreover, by comparing the three mapping results for Years 1980, 2000, and 2020, it can be observed that the Asia region in the deformed earth models becomes larger and larger from Year 1980 to 2020. This reflects the change in demographics over the past several decades, with the overall human population in Asia increasing more rapidly than in North America and South America.

In Figure 19, we show another set of spherical density-equalizing maps computed based on the nominal gross domestic product (GDP) of different regions (Asia, Africa, North America, South America, etc.) in Years 1980, 2000, and 2020, respectively. This time, we can see that North America is enlarged in all three mapping results, which shows that the nominal GDP of North America has remained relatively high over the past several decades. By contrast, the size of the Asia region shows an increasing trend in the three mapping results from Year 1980 to 2020, which reflects that economic growth in Asia has been emerging over the years. The two sets of examples above show that our method can be effectively applied to sociological data visualization.

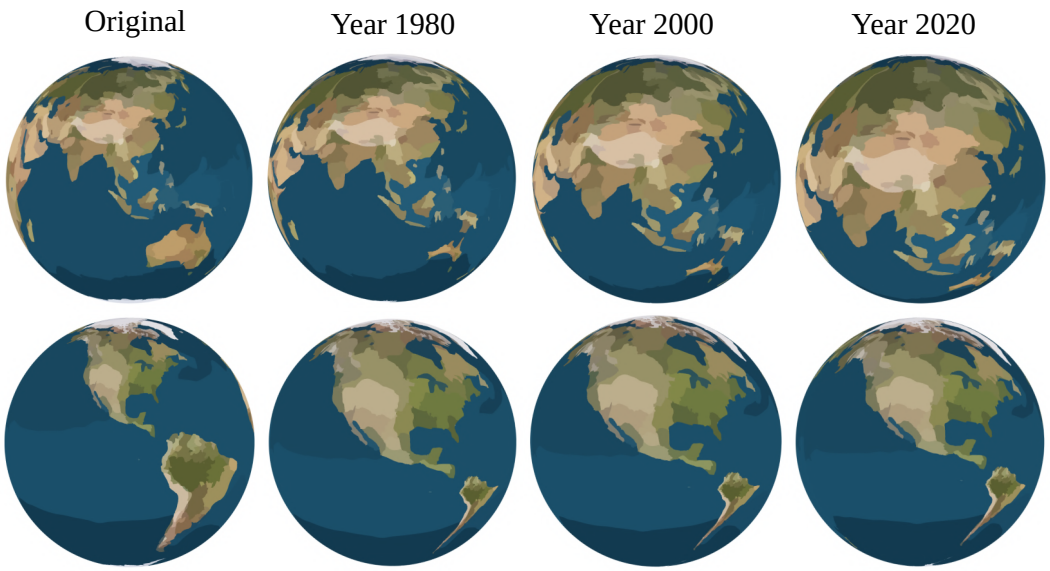

Figure 19. Visualizing the change in the nominal gross domestic product (GDP) via SDEM. *The first column shows the undeformed earth model. The second, third, and fourth columns show the deformed earth models produced by SDEM based on the nominal GDP of different regions in Years* 1980, 2000, *and* 2020, *respectively. Two different views are displayed for each earth model (top row and bottom row).*

6. Discussion. In this work, we have proposed a novel method for computing spherical density-equalizing maps (SDEM) for genus-0 closed surfaces. Using the proposed method, we can easily achieve different mapping effects, with a particular example being spherical area-preserving parameterizations. We have further proposed a combined energy model including density gradient, harmonic energy, and landmark mismatch energy to achieve landmark-aligned spherical density-equalizing maps (LSDEM) balancing different distortions. The experimental results and applications presented have demonstrated the effectiveness of our proposed methods.

We remark that while our experimental results have demonstrated the convergence of our proposed method numerically, several key steps in our algorithm pose some technical challenges in the theoretical convergence analysis. Specifically, note that while the deformation process in our proposed method is primarily guided by density diffusion solved using a semidiscrete backward Euler scheme in (3.9), our algorithm involves an additional overlap correction scheme for correcting mesh fold-overs and a recoupling scheme for recalculating the density based on the final updated vertex positions at each iteration. The effect of these schemes on the subsequent density highly depends on the input surface geometry and mesh quality. This poses challenges in establishing a theoretical convergence proof for our algorithm.

Note that both of the two proposed mapping methods are limited to surfaces with spherical topology. In the future, we plan to extend the methods for surfaces with other topologies. In particular, as many real-world surfaces are high-genus, having both a density-equalizing mapping method and a landmark-aligned density-equalizing mapping method for them would facilitate their processing and shape analysis. Also, note that the proposed mapping methods in the current work are only applicable to triangle mesh data. Another possible future direction is to extend the proposed methods for point cloud surfaces.

Data availability. The code and data are available on GitHub at https://github.com/garyptchoi/spherical-density-equalizing-map.

Conflict of interest. The authors declare no conflict of interest.

REFERENCES

[1] L. V. Ahlfors, *Lectures on Quasiconformal Mappings*, Univ. Lecture Ser. 38, American Mathematical Society, Providence, RI, 2006.

[2] S. Angenent, S. Haker, A. Tannenbaum, and R. Kikinis, *On area preserving mappings of minimal distortion*, in System Theory: Modeling, Analysis and Control, Kluwer Academic Publishers, Boston, MA, 2000, pp. 275–286.

[3] X. Chen, H. He, G. Zou, X. Zhang, X. Gu, and J. Hua, *Ricci flow-based spherical parameterization and surface registration*, Comput. Vis. Image Underst., 117 (2013), pp. 1107–1118.

[4] G. P. T. Choi, B. Chiu, and C. H. Rycroft, *Area-preserving mapping of 3D carotid ultrasound images using density-equalizing reference map*, IEEE Trans. Biomed. Eng., 67 (2020), pp. 1507–1517.

[5] G. P.-T. Choi, K. T. Ho, and L. M. Lui, *Spherical conformal parameterization of genus-0 point clouds for meshing*, SIAM J. Imaging Sci., 9 (2016), pp. 1582–1618, https://doi.org/10.1137/15M1037561.

[6] G. P. T. Choi, Y. Leung-Liu, X. Gu, and L. M. Lui, *Parallelizable global conformal parameterization of simply-connected surfaces via partial welding*, SIAM J. Imaging Sci., 13 (2020), pp. 1049–1083, https://doi.org/10.1137/19M125337X.

[7] G. P.-T. Choi, M. H.-Y. Man, and L. M. Lui, *Fast spherical quasiconformal parameterization of genus-0 closed surfaces with application to adaptive remeshing*, Geom. Imaging Comput., 3 (2016), pp. 1–29.

[8] G. P. T. Choi and C. H. Rycroft, *Density-equalizing maps for simply connected open surfaces*, SIAM J. Imaging Sci., 11 (2018), pp. 1134–1178, https://doi.org/10.1137/17M1124796.

[9] G. P. T. Choi and C. H. Rycroft, *Volumetric density-equalizing reference maps with applications*, J. Sci. Comput., 86 (2021), 41.

[10] P. T. Choi, K. C. Lam, and L. M. Lui, *FLASH: Fast landmark aligned spherical harmonic parameterization for genus-0 closed brain surfaces*, SIAM J. Imaging Sci., 8 (2015), pp. 67–94, https://doi.org/10.1137/130950008.

[11] K. Crane, U. Pinkall, and P. Schröder, *Robust fairing via conformal curvature flow*, ACM Trans. Graphics, 32 (2013), 61.

[12] L. Cui, X. Qi, C. Wen, N. Lei, X. Li, M. Zhang, and X. Gu, *Spherical optimal transportation*, Comput. Aided Des., 115 (2019), pp. 181–193.

[13] P. J. Dodd, C. Sismanidis, and J. A. Seddon, *Global burden of drug-resistant tuberculosis in children: A mathematical modelling study*, Lancet Infect. Dis., 16 (2016), pp. 1193–1201.

[14] M. T. Gastner, *Spatial Distributions: Density-Equalizing Map Projections, Facility Location, and Two-Dimensional Networks*, Ph.D. thesis, University of Michigan, Ann Arbor, MI, 2005.

[15] M. T. Gastner and M. E. J. Newman, *Diffusion-based method for producing density-equalizing maps*, Proc. Natl. Acad. Sci. USA, 101 (2004), pp. 7499–7504.

[16] M. T. Gastner, V. Seguy, and P. More, *Fast flow-based algorithm for creating density-equalizing map projections*, Proc. Natl. Acad. Sci. USA, 115 (2018), pp. E2156–E2164.

[17] R. W. Glynn, C. Scutaru, M. J. Kerin, and K. J. Sweeney, *Breast cancer research output,* 1945–2008*: A bibliometric and density-equalizing analysis*, Breast Cancer Res., 12 (2010), R108.

[18] X. Gu, Y. Wang, T. F. Chan, P. M. Thompson, and S.-T. Yau, *Genus zero surface conformal mapping and its application to brain surface mapping*, IEEE Trans. Med. Imaging, 23 (2004), pp. 949–958.

[19] A. Jacobson, *Common 3D test models*, https://github.com/alecjacobson/common-3d-test-models, 2023, Accessed: 2023-10-03.

[20] O. Lehto and K. I. Virtanen, *Quasiconformal Mappings in the Plane*, Grundlehren Math. Wiss. 126, Springer, Berlin, Heidelberg, 1973.

[21] Z. Li and S. Aryana, *Diffusion-based cartogram on spheres*, Cartogr. Geogr. Inf. Sci., 45 (2018), pp. 464–475.

[22] Z. Li and S. A. Aryana, *Visualization of subsurface data using three-dimensional cartograms*, in Advances in Remote Sensing and Geo Informatics Applications: Proceedings of the 1st Springer Conference of the Arabian Journal of Geosciences (CAJG-1), 2018, Springer, Tunisia, 2019, pp. 17–19.
[23] W.-H. Liao, T.-M. Huang, W.-W. Lin, and M.-H. Yueh, *Convergence analysis of Dirichlet energy minimization for spherical conformal parameterizations*, J. Sci. Comput., 98 (2024), 29.
[24] L. M. Lui, K. C. Lam, T. W. Wong, and X. Gu, *Texture map and video compression using Beltrami representation*, SIAM J. Imaging Sci., 6 (2013), pp. 1880–1902, https://doi.org/10.1137/120866129.
[25] L. M. Lui, Y. Wang, T. F. Chan, and P. Thompson, *Landmark constrained genus zero surface conformal mapping and its application to brain mapping research*, Appl. Numer. Math., 57 (2007), pp. 847–858.
[26] Z. Lyu, Q. Chen, and L. M. Lui, *A two-stage algorithm for combined quasiconformal and optimal mass transportation spherical parameterization*, Math. Comput. Geom. Data, 3 (2023), pp. 29–57.
[27] Z. Lyu, G. P. T. Choi, and L. M. Lui, *Bijective density-equalizing quasiconformal map for multiply connected open surfaces*, SIAM J. Imaging Sci., 17 (2024), pp. 706–755, https://doi.org/10.1137/23M1594376.
[28] H. D. Matthews, T. L. Graham, S. Keverian, C. Lamontagne, D. Seto, and T. J. Smith, *National contributions to observed global warming*, Environ. Res. Lett., 9 (2014), 014010.
[29] S. Nadeem, Z. Su, W. Zeng, A. Kaufman, and X. Gu, *Spherical parameterization balancing angle and area distortions*, IEEE Trans. Vis. Comput. Graph., 23 (2016), pp. 1663–1676.
[30] S. Nusrat and S. Kobourov, *The state of the art in cartograms*, Comput. Graph. Forum, 35 (2016), pp. 619–642.
[31] P.-O. Persson and G. Strang, *A simple mesh generator in MATLAB*, SIAM Rev., 46 (2004), pp. 329–345, https://doi.org/10.1137/S0036144503429121.
[32] U. Pinkall and K. Polthier, *Computing discrete minimal surfaces and their conjugates*, Exp. Math., 2 (1993), pp. 15–36.
[33] M. Pratt, O. L. Sarmiento, F. Montes, D. Ogilvie, B. H. Marcus, L. G. Perez, R. C. Brownson, and the Lancet Physical Activity Series Working Group, *The implications of megatrends in information and communication technology and transportation for changes in global physical activity*, Lancet, 380 (2012), pp. 282–293.
[34] A. Pumarola, J. Sanchez-Riera, G. P. T. Choi, A. Sanfeliu, and F. Moreno-Noguer, *3DPeople: Modeling the geometry of dressed humans*, in Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 2242–2251.
[35] Z. Su, W. Zeng, R. Shi, Y. Wang, J. Sun, and X. Gu, *Area preserving brain mapping*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2235–2242.
[36] W. Tobler, *Thirty five years of computer cartograms*, Ann. Am. Assoc. Geogr., 94 (2004), pp. 58–73.
[37] D. B. Wake and V. T. Vredenburg, *Are we in the midst of the sixth mass extinction? A view from the world of amphibians*, Proc. Natl. Acad. Sci. USA, 105 (2008), pp. 11466–11473.
[38] C. Wang, X. Hu, X. Fu, and L. Liu, *Bijective spherical parametrization with low distortion*, Comput. Graph., 58 (2016), pp. 161–171.
[39] Z. Wang, Z. Luo, J. Zhang, and E. Saucan, *A novel local/global approach to spherical parameterization*, J. Comput. Appl. Math., 329 (2018), pp. 294–306.
[40] M.-H. Yueh, T. Li, W.-W. Lin, and S.-T. Yau, *A novel algorithm for volume-preserving parameterizations of 3-manifolds*, SIAM J. Imaging Sci., 12 (2019), pp. 1071–1098, https://doi.org/10.1137/18M1201184.
[41] M.-H. Yueh, W.-W. Lin, C.-T. Wu, and S.-T. Yau, *An efficient energy minimization for conformal parameterizations*, J. Sci. Comput., 73 (2017), pp. 203–227.
[42] G. Zou, J. Hu, X. Gu, and J. Hua, *Authalic parameterization of general surfaces using Lie advection*, IEEE Trans. Vis. Comput. Graph., 17 (2011), pp. 2005–2014.