

# Landmark-Matching Transformation with Large Deformation Via $n$ -dimensional Quasi-conformal Maps

Yin Tat Lee<sup>1</sup> · Ka Chun Lam<sup>2</sup> · Lok Ming Lui<sup>3</sup>

Received: 21 August 2014 / Revised: 18 September 2015 / Accepted: 19 September 2015 /  
Published online: 25 September 2015  
© Springer Science+Business Media New York 2015

**Abstract** We propose a new method to obtain landmark-matching transformations between  $n$ -dimensional Euclidean spaces with large deformations. Given a set of feature correspondences, our algorithm searches for an optimal folding-free mapping that satisfies the prescribed landmark constraints. The standard conformality distortion defined for mappings between 2-dimensional spaces is first generalized to the  $n$ -dimensional conformality distortion  $K(f)$  for a mapping  $f$  between  $n$ -dimensional Euclidean spaces ( $n \geq 3$ ). We then propose a variational model involving  $K(f)$  to tackle the landmark-matching problem in higher dimensional spaces. The generalized conformality term  $K(f)$  enforces the bijectivity of the optimized mapping and minimizes its local geometric distortions even with large deformations. Another challenge is the high computational cost of the proposed model. To tackle this, we have also proposed a numerical method to solve the optimization problem more efficiently. Alternating direction method with multiplier is applied to split the optimization problem into two subproblems. Preconditioned conjugate gradient method with multi-grid preconditioner is applied to solve one of the sub-problems, while a fixed-point iteration is proposed to solve another subproblem. Experiments have been carried out on both synthetic examples and lung CT images to compute the diffeomorphic landmark-matching transformation with different landmark constraints. Results show the efficacy of our proposed model to obtain a folding-free landmark-matching transformation between  $n$ -dimensional spaces with large deformations.

---

✉ Lok Ming Lui  
lmlui@math.cuhk.edu.hk

Yin Tat Lee  
yintat@math.mit.edu

Ka Chun Lam  
kclam@math.cuhk.edu.hk

<sup>1</sup> Room E18-301A, Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA, USA

<sup>2</sup> Lady Shaw Building Room 222B, Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong

<sup>3</sup> Lady Shaw Building Room 207, Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong

**Keywords** Large deformation registration ·  $n$ -D quasi-conformal · Conformality · Alternating direction method of multipliers · Landmarks

## 1 Introduction

Finding an optimal transformation between corresponding data, such as images or geometric shapes, is an important task in various fields, such as computer vision [61], computer graphics [6, 22, 24], video processing [38, 45, 59, 60] and medical imaging [1, 25, 28, 29, 32, 49, 55]. Such a process is called *registration*. For example, in neuroimaging, it is often required to align medical images from different modalities, such as magnetic resonance (MR), X-ray computed tomography (CT) images and so on. In computer graphics, registration is necessary for texture mapping [22, 24]. Due to its important applications in different areas, an enormous amount of research has been carried out to develop effective models for registration.

Registration methods can mainly be divided into three categories, namely, (1) intensity-based registration, (2) landmark-based registration and (3) hybrid registration using both intensity and landmark information. Intensity-based registration computes a transformation between corresponding data by matching intensity functions, such as image intensity for image registration or surface curvature for surface geometric registration. Different intensity-based registration algorithms have been recently proposed [61], such as Demons [51, 56], spherical Demons [58], elastic registration [23], Large Deformation diffeomorphic Metric Mapping (LDDMM) frameworks [9, 10] and so on. On the other hand, landmark-based registration computes a smooth 1-1 dense pointwise correspondence between corresponding data that matches important features [2, 14–16, 28, 39–41, 52, 54, 57]. Such a feature-based registration approach usually comprises of two steps, namely, (1) the extraction of corresponding feature landmarks and (2) the computation of a transformation between the data that matches corresponding features. The main advantage of the landmark-based method is that intuitive user-interaction can be incorporated to guide the registration process. Recently, hybrid registration that combines landmark-based and intensity-based methods have also gained increased attention. Hybrid approaches use both the landmark and intensity information to guide the registration. This type of approaches can usually obtain more accurate registration results, since the advantages of landmark-based and intensity-based registration can be combined. Different hybrid registration models have also been proposed recently [7, 17, 26, 27, 44].

In this work, we will focus on the landmark-based registration. Landmark-based registration has found important applications. One typical example is the brain cortical surface registration for which sulcal landmarks are usually extracted to guide the registration [41, 52, 57]. Landmark-based registration has also been applied to register gene expression data to a neuroanatomical mouse atlas [36]. Feature-matching image registration can also be used as an initial guess for intensity-based registration between images with large deformations [27, 33]. Over the past few decades, numerous landmark-based registration models have been proposed [2, 21, 28, 31, 47]. One of the first and most important landmark-based registration algorithm is the Thin-Plate Spline (TPS) method proposed by Bookstein [2]. TPS minimizes the bending energy together with the landmark mismatching term. A unique and closed-form solution is guaranteed in this model. TPS is efficient and works well under small deformations. However, under larger deformations, TPS generally cannot preserve the bijectivity of the mapping [11].

In some situations like medical image registration or constrained texture mapping of surfaces, a bijective and topology-preserving mapping is desirable for the registration problem

[49]. For example, Christensen et al. [9] proposed a regridding algorithm to restrict the transformation of the image deformation to have a globally positive definite Jacobian. Statistically, Leow et al. [35] studied the statistical properties of Jacobian maps (the determinant of the Jacobian matrix of a deformation field) and proposed a framework for constructing unbiased deformation fields. Modat et al. [43] also proposed a variational model with the joint bending energy and the squared Jacobian determinant penalty terms to obtain a transformation for lung registration.

Recently, quasi-conformal (QC) theory has been introduced to handle large deformation landmark-matching registration problem [30,33,37,59]. The Beltrami coefficient, which measures the conformality distortion, can be effectively used to enforce the bijectivity of the mapping. By optimizing an energy functional involving the  $L^p$ -norm of the Beltrami coefficient, large deformation diffeomorphic registration can be accurately computed. Several works have also been proposed to deal with surface-based landmark-matching problem with different genus [32,42]. QC theory has provided an effective framework to handle registration problem with large deformations for 2-dimensional spaces. However, for general  $n$ -dimensional spaces, the conformality distortion is not defined. Motivated by this, it is our goal in this paper to extend the concept of 2D quasi-conformality to general  $n$ -dimensional spaces. In particular, a notion of conformality distortion of a diffeomorphism in the  $n$ -dimensional Euclidean space will be formulated. With the definition of conformality distortion, we can extend the 2D quasi-conformal registration algorithm to general  $n$ -dimensional Euclidean spaces.

In short, the main contributions in this paper are threefolded:

1. We give a definition of  $n$ -dimensional conformality distortion  $K(f)$  ( $n \geq 3$ ) for mapping  $f$  between  $n$ -dimensional Euclidean spaces. The standard conformality distortion is defined for 2-dimensional space. Our definition aims to generalize this concept to  $n$ -dimensional spaces.
2. With the definition of  $K(f)$ , we extend our previous model [33] for computing 2-dimensional landmark-matching bijective mapping with large deformations to higher dimensional spaces. This allows us to compute bijective landmark-matching mapping of higher dimensional spaces with large deformations.
3. One challenge of the proposed model is the high computational cost for higher-dimensional spaces. In this paper, we propose a numerical method to solve the optimization problem more efficiently. This is based on applying the alternating direction method with multiplier (ADMM) to split the problem into two subproblems. Preconditioned conjugate gradient method with multi-grid V-cycle preconditioner is applied to solve one of the subproblems. A fixed-point iteration is proposed to solve another subproblem, whose convergence to the minimizer is theoretically shown.

This paper is organized as follows. In Sect. 2, basic mathematical background will be explained. In Sect. 3, we describe our proposed model to obtain the landmark-matching transformation with large deformation between  $n$ -dimensional Euclidean spaces in details. The numerical algorithm will be discussed in Sect. 4. Experimental results will be demonstrated in Sect. 5. Conclusion and future work will be discussed in Sect. 6.

## 2 Mathematical Background

In this section, we describe some basic mathematical concepts related to our algorithms. For details, we refer the readers to [12,34].

A surface  $S$  with a conformal structure is called a *Riemann surface*. Given two Riemann surfaces  $M$  and  $N$ , a map  $f : M \rightarrow N$  is *conformal* if it preserves the surface metric up to a scalar multiplicative factor called the conformal factor. An immediate consequence is that every conformal map preserves angles. With the angle-preserving property, a conformal map effectively preserves the local geometry of the surface structure.

A generalization of conformal maps is the *quasi-conformal* maps, which are orientation preserving homeomorphisms between Riemann surfaces with bounded conformality distortion, in the sense that their first order approximations takes small circles to small ellipses of bounded eccentricity [12]. Surface registrations and parameterizations, which are orientation-preserving homeomorphisms, can be considered as quasi-conformal maps. Mathematically,  $f : \mathbb{C} \rightarrow \mathbb{C}$  is quasi-conformal provided that it satisfies the Beltrami equation:

$$\frac{\partial f}{\partial \bar{z}} = \mu(z) \frac{\partial f}{\partial z}. \quad (1)$$

for some complex valued Lebesgue measurable  $\mu$  satisfying  $\|\mu\|_\infty < 1$ .  $\mu$  is called the *Beltrami coefficient*, which is a measure of non-conformality. In particular, the map  $f$  is conformal around a small neighborhood of  $p$  when  $\mu(p) = 0$ . Infinitesimally, around a point  $p$ ,  $f$  may be expressed with respect to its local parameter as follows:

$$\begin{aligned} f(z) &\approx f(p) + f_z(p)z + f_{\bar{z}}(p)\bar{z} \\ &= f(p) + f_z(p)(z + \mu(p)\bar{z}). \end{aligned} \quad (2)$$

Obviously,  $f$  is not conformal if and only if  $\mu(p) \neq 0$ . Inside the local parameter domain,  $f$  may be considered as a map composed of a translation to  $f(p)$  together with a stretch map  $S(z) = z + \mu(p)\bar{z}$ , which is postcomposed by a multiplication of  $f_z(p)$ , which is conformal. All the conformality distortion of  $S(z)$  is caused by  $\mu(p)$ .  $S(z)$  is the map that causes  $f$  to map a small circle to a small ellipse. From  $\mu(p)$ , we can determine the angles of the directions of maximal magnification and shrinking and the amount of them as well. Specifically, the angle of maximal magnification is  $\arg(\mu(p))/2$  with magnifying factor  $1 + |\mu(p)|$ ; The angle of maximal shrinking is the orthogonal angle  $(\arg(\mu(p)) - \pi)/2$  with shrinking factor  $1 - |\mu(p)|$ . The distortion or dilation is given by:

$$K = (1 + |\mu(p)|) / (1 - |\mu(p)|). \quad (3)$$

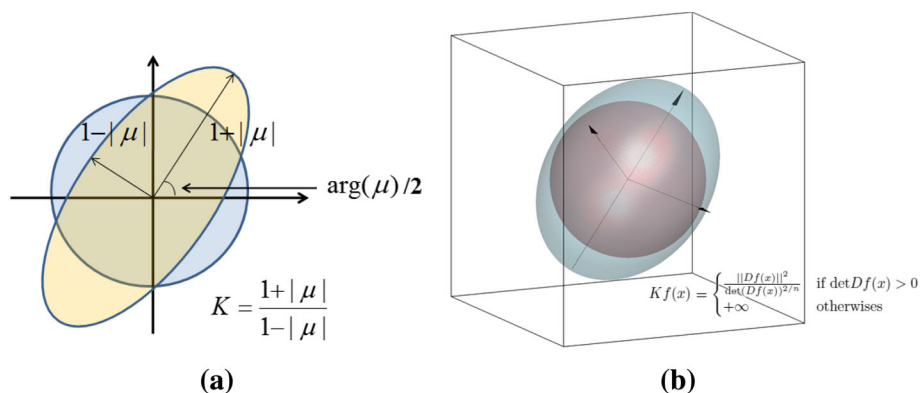
Thus, the Beltrami coefficient  $\mu$  gives us all the information about the properties of the map (See Fig. 1a).

Given a Beltrami coefficient  $\mu : \mathbb{C} \rightarrow \mathbb{C}$  with  $\|\mu\|_\infty < 1$ . There is always a quasi-conformal mapping from  $\mathbb{C}$  onto itself which satisfies the Beltrami equation in the distribution sense [12].

However, the above quasi-conformal theories only apply to two dimensional spaces or surfaces. In this work, our goal is to extend the idea of 2-dimensional quasi-conformal theories to general  $n$ -dimensional spaces. We will introduce a notion of conformality distortion of a diffeomorphism of the  $n$ -dimensional Euclidean space. The conformality distortion measures the distortion of an infinitesimal ball to an infinitesimal ellipsoid under the diffeomorphism (See Fig. 1b).

### 3 Proposed Model

In this section, we will explain in details our proposed model to obtain the landmark-matching transformation between  $n$ -dimensional Euclidean spaces. The basic idea is to formulate the



**Fig. 1** Illustration of the conformality distortion. **a** Shows how a *small circle* is deformed to an *ellipse* under a 2D quasi-conformal map. The conformality distortion is measured by the Beltrami coefficient, **b** Shows how a *small ball* is deformed to a *small ellipsoid* under a 3D diffeomorphism. The conformality distortion can be measured by  $K(f)$  defined in this paper

notion of conformality distortion of a diffeomorphism of the  $n$ -dimensional Euclidean space. The conformality distortion measures the distortion of an infinitesimal ball to an infinitesimal ellipsoid under the diffeomorphism. The landmark-matching problem can then be modelled as minimizing an energy functional involving a conformality term and a smoothness term under the prescribed landmark constraints. We first introduce the conformality distortion of a diffeomorphism of the  $n$ -dimensional Euclidean space. In Sect. 3.2, we describe the continuous model of the proposed energy functional. Finally, we explain the discretization of the model in Sect. 3.3.

### 3.1 Conformality Distortion

Let  $\Omega_1, \Omega_2 \subset \mathbb{R}^n$  be the domain and the image of the diffeomorphism  $f = (f_1, f_2, \dots, f_n) : \Omega_1 \rightarrow \Omega_2$  respectively. For any  $\mathbf{p} = (p_1, p_2, \dots, p_n) \in \Omega_1$ , let  $\mathbf{q} = (q_1, q_2, \dots, q_n) = f(\mathbf{p})$ . Then, for any  $\mathbf{x} = (x_1, \dots, x_n)$  in a neighbourhood of  $\mathbf{p}$ , we have

$$\mathbf{y} = f(\mathbf{x}) \approx f(\mathbf{p}) + Df(\mathbf{x} - \mathbf{p}), \quad (4)$$

where  $Df = (\frac{\partial f_i}{\partial x_j})_{1 \leq i, j \leq n} \in M_{n \times n}(\mathbb{R})$ .

Under a general diffeomorphism  $f$ ,  $f$  distorts an infinitesimal ball  $B_\epsilon(\mathbf{p}) := \{\mathbf{x} \in \Omega_1 : \|\mathbf{x} - \mathbf{p}\| \leq \epsilon\}$  to an infinitesimal ellipsoid  $E_f$  (see Fig. 1b). More precisely,

$$\begin{aligned} E_f &= \{\mathbf{q} + Df(\mathbf{x} - \mathbf{p}) : \mathbf{x} \in B_\epsilon(\mathbf{p})\} \\ &= \{\mathbf{q} + \mathbf{w} : \mathbf{w}^T C \mathbf{w} \leq \epsilon\}, \end{aligned} \quad (5)$$

where  $C = ((Df)^{-1})^T (Df)^{-1}$  is a symmetric positive definite matrix. Obviously, since  $C$  is symmetric positive definite,  $E_f$  is an ellipsoid centered at  $\mathbf{q}$ . Moreover,  $E_f$  is an infinitesimal ball if all eigenvalues of  $C$  are equal. This can be observed easily as follows. Suppose  $C = Q^T DQ$ , where  $Q$  is an orthogonal matrix and  $D$  is a diagonal matrix consisting of the eigenvalues of  $A$ . If  $D = \mu \mathbf{I}$  ( $\mu > 0$ ),  $C = \mu \mathbf{I}$ . It follows that for any  $\mathbf{y} \in E_f$ ,  $(\mathbf{y} - \mathbf{q})^T C (\mathbf{y} - \mathbf{q}) = \mu (\mathbf{y} - \mathbf{q})^T (\mathbf{y} - \mathbf{q}) \leq \epsilon$ . This gives  $(\mathbf{y} - \mathbf{q})^T (\mathbf{y} - \mathbf{q}) \leq \epsilon/\mu$ . Hence,  $E_f$  is an infinitesimal ball with radius  $\epsilon/\mu$ .

To define the conformality distortion, we define a measurement that quantifies the geometric distortion of the ellipsoid  $E_f$  from an infinitesimal ball. From the above observation, it is the same as measuring how far the matrix  $C$  is from a symmetric positive definite matrix with equal eigenvalues. It is related to the Jacobian of the mapping  $f$ .

Consider  $A = (Df)^T(Df)$ . The eigenvalues of  $C$  are equal if and only if the eigenvalues of  $A = (Df)^T(Df)$  are equal. Suppose  $\lambda_1, \lambda_2, \dots, \lambda_n$  are the eigenvalues of  $A$ . Using the AM–GM inequality, we have

$$(\lambda_1 \cdots \lambda_n)^{1/n} \leq \frac{\lambda_1 + \cdots + \lambda_n}{n} \quad \text{or,} \\ \frac{1}{n} \left( \frac{\lambda_1 + \cdots + \lambda_n}{(\lambda_1 \cdots \lambda_n)^{1/n}} \right) \geq 1 \quad (6)$$

where the equality sign holds if and only if  $\lambda_1 = \cdots = \lambda_n$ . Hence, we can define the  $n$ -D conformality distortion as  $K(f) = \frac{1}{n} \left( \frac{\lambda_1 + \cdots + \lambda_n}{(\lambda_1 \cdots \lambda_n)^{1/n}} \right)$ . Note that the AM–GM inequality states that the perimeter of the  $n$ -dimensional cubes is the smallest amongst all  $n$ -dimensional rectangular boxes with the same volume. In the 2-dimensional case, the perimeter of a square is always the smallest amongst all rectangles with a given area. Hence,  $K(f)$  can be interpreted as the ratio of the perimeter of a  $n$ -dimensional rectangular boxes with edges lengths equal to the eigenvalues of  $A$  to the perimeter of the  $n$ -cube with the same volume. Let  $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$  be the orthonormal basis of eigenvectors of  $A$ . Then, the minimum is attained if the  $n$ -dimensional boxes spanned by  $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$  is a  $n$ -cube. This happens when all eigenvalues are equal.

Now, the arithmetic mean and geometric mean of the eigenvalues of  $A$  can be expressed as the Frobenius norm and determinant of  $Df$  respectively. Observe that:

$$\|Df\|_F^2 = \text{Tr}(Df^T Df) = \text{Tr}(A) = \lambda_1 + \cdots + \lambda_n; \\ \det(A) = \lambda_1 \cdots \lambda_n = \det(Df^T Df) = \det(Df)^2, \quad (7)$$

where  $\|Df(x)\|_F$  denotes the Frobenius norm of  $Df(x)$ . Therefore, we can now introduce the following definition:

**Definition 1** (Conformality distortion) The conformality distortion  $Kf(x)$  of a mapping  $f$  at point  $x$  is defined by

$$Kf(x) := \begin{cases} \frac{1}{n} \left( \frac{\|Df(x)\|_F^2}{\det(Df(x))^{2/n}} \right) & \text{if } \det(Df(x)) > 0, \\ +\infty & \text{otherwise} \end{cases} \quad (8)$$

Note that  $Kf(x) \geq 1$  and  $Kf(x) = 1$  if and only if  $E_{\mathbf{x}} = \{\mathbf{x} + Df(\mathbf{y} - \mathbf{x}) : \mathbf{y} \in B_\epsilon(\mathbf{x})\}$  is a  $n$ -dimensional ball. Motivated by this observation, we say  $f$  is conformal at point  $x$  if the conformality distortion  $Kf(x)$  attains its minimum value 1. By setting  $Kf(x) = +\infty$  when  $\det(Df(x)) \leq 0$ , we can ensure the bijectivity of the mapping by minimizing the norm of  $Kf(x)$ .

For  $n = 2$ , denote  $f(x_1, x_2) = f_1(x_1, x_2) + \sqrt{-1}f_2(x_1, x_2)$  and assume  $\det(Df(x)) > 0$  for all  $x$ . Then,  $\frac{\partial f}{\partial \bar{z}} = \left( \frac{\partial f_1}{\partial x_1} + \frac{\partial f_2}{\partial x_2} \right) + \sqrt{-1} \left( \frac{\partial f_2}{\partial x_1} - \frac{\partial f_1}{\partial x_2} \right)$  and  $\frac{\partial f}{\partial z} = \left( \frac{\partial f_1}{\partial x_1} - \frac{\partial f_2}{\partial x_2} \right) + \sqrt{-1} \left( \frac{\partial f_2}{\partial x_1} + \frac{\partial f_1}{\partial x_2} \right)$ . We have

$$\begin{aligned}
Kf(x) &= \frac{1}{2} \frac{\|Df(x)\|_F^2}{\det(Df(x))} = \frac{\left(\frac{\partial f_1}{\partial x_1}\right)^2 + \left(\frac{\partial f_1}{\partial x_2}\right)^2 + \left(\frac{\partial f_2}{\partial x_1}\right)^2 + \left(\frac{\partial f_2}{\partial x_2}\right)^2}{2\det(Df(x))} \\
&= \frac{|f_z|^2(1 + |\mu|^2)}{|f_z|^2(1 - |\mu|^2)} \\
&= \frac{1 + |\mu|^2}{1 - |\mu|^2}
\end{aligned} \tag{9}$$

where  $\mu(x)$  is the Beltrami coefficient defined in Eq. (1).

### 3.2 The Continuous Model

With the notion of  $n$ -D conformality distortion  $K(f)$ , we can now develop a variational model to compute a landmark-matching transformation between  $n$ -dimensional spaces. Given two domains  $\Omega_1$  and  $\Omega_2$  in  $\mathbb{R}^n$ . Suppose  $\{p_i \in \Omega_1\}_{i=1}^m$   $\{q_i \in \Omega_2\}_{i=1}^m$  are corresponding feature landmarks in  $\Omega_1$  and  $\Omega_2$  respectively. These corresponding sets of feature landmarks gives the landmark constraints on the mapping. Our goal is to search for a bijective transformation  $f : \Omega_1 \rightarrow \Omega_2$  that satisfies  $f(p_i) = q_i$  for  $i = 1, 2, \dots, m$ . In other words, the selected landmark points  $p_i$  in  $\Omega_1$  are required to be mapped to the corresponding  $q_i$  in  $\Omega_2$ . This is called the landmark-based registration problem. Most of the existing landmark-based registration models are variational approaches. They can mainly be written as minimizing:

$$E(f) = \text{Reg}(f) \text{ subject to: } f(p_i) = q_i \text{ for } i = 1, \dots, n \text{ (hard landmark constraints),} \tag{10}$$

where  $\text{Reg}(f)$  is the regularization of the mapping  $f$ . Sometimes, the hard landmark constraints can be relaxed by minimizing

$$E(f) = \text{Reg}(f) + \lambda \sum_{i=1}^n \|f(p_i) - q_i\|_2^2 \text{ (soft landmark constraints)} \tag{11}$$

For example, the popular thin-plate spline (TPS) landmark-based registration model uses the integral of the square of the second derivative as the regularizer [2]. In this work, we propose to use the  $L^1$ -norm of the conformality distortion  $K(f)$  together with a smoothness term  $\|\Delta f\|_2^2$  as the regularizer. Note that the conformality distortion has also been used to obtain registration for 2-dimensional spaces (such as 2D images or 2D surfaces) [33]. In this paper, with the notion of  $n$ -D conformality distortion  $Kf(x)$ , we extend this idea to general  $n$ -dimensional spaces. This allows us to compute bijective landmark-matching mappings of higher dimensional spaces with large deformations. Another challenge is the high computational cost of the proposed model in the high dimensional space. To handle this, we also propose a numerical method to solve the optimization problem more efficiently (please refer to Sect. 4).

With conformality distortion  $Kf(x)$ , the bijectivity of the registration can be easily guaranteed by enforcing the constraint  $\|Kf(x)\|_\infty < K$  for some  $K < \infty$ . This can be achieved by minimizing an energy functional involving  $\|Kf(x)\|_\infty$ . In addition, minimizing  $\|Kf(x)\|_\infty$  also helps to reduce the maximal conformality distortion, and hence reduce the local geometric distortion of the mapping. However, it is computationally expensive to minimize an energy functional involving the supremum norm. Consequently, we propose to minimize  $\|Kf(x)\|_1$ . Since  $Kf(x)$  is set to be  $+\infty$  when  $\det(Df(x)) \leq 0$ , our variational model can still prevent folding by minimizing  $\|Kf(x)\|_1$ .



Besides,  $\|Df\|_F$  is included in the term  $Kf$ , which is the commonly used squared Frobenius regularization term. The smoothness of the mapping can be achieved by merely minimizing the conformality term. In order to further enhance the smoothness of the mapping, an extra smoothness term can be included in the energy functional. We now propose the minimization model for the landmark-based registration problem as follows:

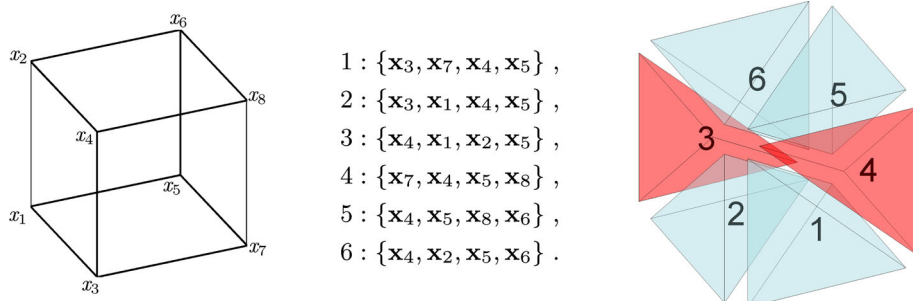
$$\inf_{f \in F} \|Kf(x)\|_1 + \frac{\sigma}{2} \|\Delta f(x)\|_2^2 dx \quad (12)$$

where  $\sigma \geq 0$  is a fixed parameter and  $F = \{f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n | f(p_i) = q_i, i = 1, 2, \dots, m\}$  is the set of functions  $f : \Omega \rightarrow \mathbb{R}^n$  which satisfies the landmark constraint  $f(p_i) = q_i$ , where  $p_i$  and  $q_i$  are the given landmark points ( $i = 1, 2, \dots, m$ ). The first energy term helps to obtain a quasi-conformal map with minimal conformality distortion, while satisfying the landmark constraints. The second energy term aims to further enhance the smoothness of the mapping, since it involves higher order derivatives. Again, since  $\|Df\|_F$  is included in  $Kf$ , some smoothness can already be achieved by minimizing the first energy term. In practice, we set  $\sigma = 0$ , which is enough to give smooth landmark-aligned mappings. This improves the efficiency of the algorithm. In extreme situations (such as a very large deformation), setting a non-zero  $\sigma$  can help to achieve much smoother registration results.

### 3.3 The Discrete Model

For general Euclidean space, our model (12) can be discretized by using discrete differential forms. For the ease of explanation, we will explain the discretization of (12) on a cubic domain in the 3-D space here. We introduce a regular tetrahedral mesh on the cubic domain as follows: We first partition the cubic domain into small equal-size cubes. In each small cube with vertices  $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8\}$ , we construct the tetrahedral subdivision of the cube with 6 tetrahedra such that each tetrahedron in the small cube contains 3 edges, each one of them is parallel to the one of the three coordinate axis respectively. More precisely, the information of these 6 tetrahedra are shown in Fig. 2.

Consider the affine map  $A$  associated with each tetrahedron. Denote  $\mathbf{u}_k = (x_k, y_k, z_k) \in \mathbb{R}^3$ ,  $k = 0, 1, 2, 3$  be the coordinates of the four vertices of the tetrahedron in Euclidean space. We also denote the image of the affine map to be  $A(\mathbf{u}_k) = \mathbf{v}_k = (\tilde{x}_k, \tilde{y}_k, \tilde{z}_k) \in \mathbb{R}^3$ . In matrix notation, we have



**Fig. 2** The six tetrahedra constructed in each unit cube with vertices  $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8\}$



$$\begin{pmatrix} \tilde{x}_0 & \tilde{x}_1 & \tilde{x}_2 & \tilde{x}_3 \\ \tilde{y}_0 & \tilde{y}_1 & \tilde{y}_2 & \tilde{y}_3 \\ \tilde{z}_0 & \tilde{z}_1 & \tilde{z}_2 & \tilde{z}_3 \\ 1 & 1 & 1 & 1 \end{pmatrix} = A \begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ z_0 & z_1 & z_2 & z_3 \\ 1 & 1 & 1 & 1 \end{pmatrix}, \quad \text{where } A = \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} \quad (13)$$

We then have

$$A = \begin{pmatrix} \tilde{x}_0 & \tilde{x}_1 & \tilde{x}_2 & \tilde{x}_3 \\ \tilde{y}_0 & \tilde{y}_1 & \tilde{y}_2 & \tilde{y}_3 \\ \tilde{z}_0 & \tilde{z}_1 & \tilde{z}_2 & \tilde{z}_3 \\ 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ z_0 & z_1 & z_2 & z_3 \\ 1 & 1 & 1 & 1 \end{pmatrix}^{-1} \quad (14)$$

Thus the Jacobian matrix of the affine map  $A$  is

$$D(A) = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix} \quad (15)$$

Denote  $Df(T)$  to be the  $3 \times 3$  Jacobian matrix of  $f$  for tetrahedron  $T$ . The discrete version of (12) is given by

$$\inf_{f \in F} \sum_{\text{tetrahedron } T} Kf(T) + \frac{\sigma}{2} \sum_{\text{node } x} \|\Delta f(x)\|_2^2 \quad (16)$$

where  $f \in F = \{f : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}^3 | f(p_i) = q_i, i = 1, 2, \dots, m\}$  is the set of functions defined on nodes of the mesh, seven-point Laplacian stencil with suitable boundary condition (which will be discussed in Sect. 4 is used for  $\Delta f$  and  $Kf(T)$  is defined by:

$$Kf(T) = \begin{cases} \frac{\|Df(T)\|_F^2}{\det(Df(T))^{2/3}} & \text{if } \det(Df(T)) > 0, \\ +\infty & \text{otherwise} \end{cases} \quad (17)$$

## 4 Algorithm

In this section, we explain the numerical algorithm to optimize the energy functional described in the last section. For the ease of the explanation, we will demonstrate the numerical algorithms for the 3-D case. The numerical algorithms for general  $n$ -D spaces can be done similarly. We split the optimization problem (16) as follows:

$$\inf_{f, R} \sum_{\text{tetrahedron } T} K(f, R, T) + \frac{\sigma}{2} \sum_{\text{node } x} \|\Delta f(x)\|_2^2 \quad \text{given } R(T) = Df(T) \quad (18)$$

where:

$$K(f, R, T) = \begin{cases} \frac{\|Df(T)\|_F^2}{\det(R(T))^{2/3}} & \text{if } \det(R(T)) > 0, \\ +\infty & \text{otherwise} \end{cases} \quad (19)$$

We apply the ADMM to optimize (19). ADMM was firstly proposed in 1975 and has received lots of attention recently due to its simple implementation and extensive applications to image processing and compressive sensing. We will briefly describe the idea of ADMM. For details, we refer the readers to [4, 18, 19]. For a general optimization problem

$$\inf_{x, y} E(x, y) \quad \text{subject to } y = Ax \quad (20)$$

where  $A \in M_{m \times n}(\mathbb{R})$ . The augmented Lagrangian associated to the problem (20) is given by

$$L(x, y, \lambda, \eta) = E(x, y) + \frac{\eta}{2} \|Ax - y - \lambda\|^2 \quad (21)$$

The ADMM decouples the optimization process, which can be described as follows:

$$\begin{aligned} x^{k+1} &= \operatorname{argmin}\{L(x, y^k, \lambda^k, \eta^k)\} \\ y^{k+1} &= \operatorname{argmin}\{L(x^{k+1}, y, \lambda^k, \eta^k)\} \\ \lambda^{k+1} &= \lambda^k + \eta_k (Ax^{k+1} - y^{k+1}) \end{aligned} \quad (22)$$

where  $\{\lambda^k\}$  is the sequence approximating the Lagrange multiplier of the constraint  $Ax = y$  and  $\{\eta_k\}$  is a sequence of positive real numbers, called the *penalty parameters*. Variants of the choices of  $\{\lambda^k\}$  and  $\{\eta_k\}$  have been proposed. In other words, ADMM firstly solves for  $x^{k+1}$  by fixing  $y = y^k$ , and then solves for  $y^{k+1}$  by fixing  $x = x^{k+1}$ . This leads to an efficient and parallelizable optimization algorithm.

Applying the ADMM to our problem, our numerical algorithm can now be described as follows. Suppose  $(f^k, R^k, \lambda_k, \eta_k)$  is obtained at the  $k$ -th iteration. We first solve the f-subproblem:

$$\begin{aligned} f^{k+1} &= \operatorname{argmin}_f \sum_T K(f, R^k, T) + \frac{\eta}{2} \sum_T \|R^k - Df + \lambda^k\|_F^2 + \frac{\sigma}{2} \sum_x \|\Delta f\|_2^2 \\ &\text{subject to } f^{k+1}(p_i) = q_i \text{ for } i = 1, \dots, m. \end{aligned} \quad (23)$$

We then solve the R-subproblem:

$$R^{k+1}(T) = \operatorname{argmin}_{\det(R) > 0} K(f^{k+1}, R, T) + \frac{\eta}{2} \|R - Df^{k+1} + \lambda^k\|_F^2 \text{ for each tetrahedron } T. \quad (24)$$

The Lagrange multiplier  $\lambda_{k+1}$  and the penalty parameter  $\eta_{k+1}$  are updated as follows.

$$\lambda^{k+1} = \lambda^k + R^{k+1} - Df^{k+1}; \quad (25)$$

$$\eta_{k+1} = \max \left\{ \max_T \frac{30}{\det(R(T))^{2/3}}, \eta_k \right\}. \quad (26)$$

Note that we vary the penalty parameter  $\eta_k$  in each iteration. The goal of varying the parameter is to improve the convergence and robustness of the algorithm in practice. As shown in [46], super-linear convergence of ADMM can usually be achieved if the penalty parameter is iteratively increased to infinity. We adopt this idea to vary the penalty parameter. The outer “max” expression in (26) ensures that  $\eta_k$  is increasing throughout the iterations. On the other hand, the choice of  $\max_T \frac{30}{\det(R(T))^{2/3}}$  is to ensure that the penalty parameter is not too big nor too small. Recall that the penalty parameter  $\eta$  in the ADMM aims to drive  $R$  to be closer to  $Df$ , so that the optimal solution eventually satisfies the constraint  $R = Df$ . If  $\eta$  is too small, the solution in each ADMM iteration may be far away from the admissible solution satisfying the constraint  $R = Df$ . It may take a long time to converge to the optimizer of (18) satisfying the constraint  $R = Df$ . On the other hand, if  $\eta$  is too big, the solution in each ADMM iteration better satisfies the constraint  $R = Df$ . But again, it may take a long time to obtain the optimizer minimizing the energy functional of the original problem (18) (since the minimization is dominated by the penalty term for the constraint). Hence, an optimal penalty parameter has to be carefully chosen [13]. From our experimental results, we found that our algorithm with the choice of  $\max_T \frac{30}{\det(R(T))^{2/3}}$  is robust for either small or large deformations.

**Algorithm 1:** *Quasi-conformal landmark-matching transformation algorithm***Input:** Domain  $\Omega$ ; landmark sets  $\{p_i \in \Omega\}_{i=1}^m$  and  $\{q_i \in \Omega\}_{i=1}^m$ .**Output:** Landmark registration  $f^* : S_1 \rightarrow S_2$ .

```

1 Initial  $f^1 = \text{Identity map } \mathbb{I}$ ;  $R^1 = Df^1$ ;  $\lambda^1 = 0$ ;
2 repeat
3    $f^{k+1} \leftarrow \underset{f}{\operatorname{argmin}} \sum_T K(f, R^k, T) + \frac{\eta}{2} \sum_T \|R^k - Df + \lambda^k\|_F^2 + \frac{\sigma}{2} \sum_x \|Df\|_2^2$ 
4   subject to  $f^{k+1}(p_i) = q_i$  for  $i = 1, \dots, m$ ;
5    $R^{k+1} \leftarrow \underset{\det(R)>0}{\operatorname{argmin}} K(f^{k+1}, R, T) + \frac{\eta}{2} \|R - Df^{k+1} + \lambda^k\|_F^2$  for each tetrahedron  $T$ ;
6    $\lambda^{k+1} \leftarrow \lambda^k + R^{k+1} - Df^{k+1}$ ;
7   Update  $\eta_{k+1} = \max\{\max_T \frac{30}{\det(R(T))^{2/3}}, \eta_k\}$ ;
8    $k \leftarrow k + 1$ ;
9 until  $\|f^{k+1} - f^k\|_\infty \leq \epsilon$ ;
```

The overall algorithm can now be summarized as Algorithm 1.

There are two subproblems in the algorithm, namely, the  $f$ -subproblem and the  $R$ -subproblem. In practice, we solve the  $f$ -subproblem first. In the following two subsections, we will explain how the  $f$ -subproblem and the  $R$ -subproblem can be solved in details.

#### 4.1 $f$ -subproblem

The  $f$ -subproblem is to minimize the energy

$$E_{fsub}^k(f) = \sum_T \frac{\|Df(T)\|_F^2}{\det(R^k(T))^{2/3}} + \frac{\eta}{2} \sum_T \|R^k(T) - Df(T) + \lambda^k(T)\|_F^2 + \frac{\sigma}{2} \sum_x \|Df\|_2^2. \quad (27)$$

Note that  $f = (f_1, f_2, f_3) : \Omega_1 \subset \mathbb{R}^3 \rightarrow \Omega_2 \subset \mathbb{R}^3$  is a vector-valued function. The energy functional  $E_{fsub}^k$  can be decoupled into  $E_{fsub}(f) = E_{fsub1}^k(f_1) + E_{fsub2}^k(f_2) + E_{fsub3}^k(f_3)$ . The optimization problem can be solved componentwisely. Therefore, we can regard  $f$  as a scalar function only in this section. The corresponding Euler-Lagrange equation for this problem is of the form

$$\begin{cases} \sigma \Delta^2 f(x) - \nabla \cdot (A(x) \nabla f(x)) = g(x); \\ f(p_i) = q_i, \end{cases} \quad (28)$$

where  $A(x)$  is a diagonal matrix with diagonal entries

$$\sum_{\text{six } T \text{ touch the corresponding edge}} \left( \frac{1}{\det(R(T))^{2/3}} + \eta \right), \quad (29)$$

and  $g(x) = -\eta \nabla \cdot (R^k(T) + \lambda^k(T))$ . To simplify the partial differential equation in (28), we substitute  $f(x) = \tilde{f}(x) + l(x)$ , where  $l(x)$  is the identity map  $\mathbb{I}$  except at the landmark points such that  $l(p_i) = q_i$  for  $i = 1, 2, \dots, m$ . By this substitution and the modification of the function  $g(x)$ , we can simplify the constraints from  $f(p_i) = q_i$  to  $\tilde{f}(p_i) = 0$ . Therefore, we may assume  $f(p_i) = 0$  in the following discussion.

To solve the linear system obtained from discretization of Eq. (28), we apply the preconditioned conjugate gradient (PCG) method as described in [48] in order to speed up the computation. The basic idea of preconditioning is to find a preconditioning matrix  $M$  such

that we can transform the original linear system into one with the same solution, but the transformed linear system is easier to solve with an iterative solver. In this work, we apply the multigrid preconditioning method to construct the transformation  $M$ . With the preconditioner  $M$ , the original  $f$ -subproblem will be solved exactly using the PCG method.

To further reduce the computation cost in applying the multigrid preconditioning method, we set  $\eta = \max_T \frac{30}{\det(R(T))^{2/3}}$  as suggested in the last section and we replace  $A(x)$  by  $A(x) = 6\eta I$ . Hence, Eq. (28) can be approximated by a Poisson equation

$$\begin{cases} \sigma \Delta^2 f(x) - 6\eta \Delta f(x) = g(x), \\ f(p_i) = 0. \end{cases} \quad (30)$$

We proceed to approximate the solution of the above approximated system to get a preconditioner  $M$ . We remark that the above approximated system is introduced to obtain the preconditioner  $M$ . With the preconditioner  $M$ , the original  $f$ -subproblem (28) will be solved exactly using the PCG method.

If  $\sigma \neq 0$ , the Eq. (30) can be split into two coupled Poisson equations

$$\begin{cases} -h - \Delta f = 0, \\ -\sigma \Delta h - 6\eta \Delta f = g, \\ f(p_i) = 0. \end{cases} \quad (31)$$

If  $\sigma = 0$ , the Eq. (30) can be simplified to be the following Poisson equation

$$\begin{cases} -6\eta \Delta f(x) = g(x), \\ f(p_i) = 0. \end{cases} \quad (32)$$

In both cases, the equation can be approximately solved using the multi-grid V-cycle, which gives us a preconditioner  $M$  for solving (28).

We will now explain the multi-grid V-cycle briefly. For details, we refer the readers to [5, 53].

Let us first define a hierarchy of discretization of the unit cube, that is  $V_1 \subset V_{2^{-1}} \subset \dots \subset V_{2^{-J}}$  where  $V_h$  is a uniform grid on unit cube with spacing  $h$ . On  $V_h$ , we discretize the Eq. (31) and (32) respectively as

$$\begin{cases} -\begin{pmatrix} I & L_h \\ L_h & 6\frac{\eta}{\sigma}L_h \end{pmatrix} \begin{pmatrix} h \\ f \end{pmatrix} = \begin{pmatrix} 0 \\ g \end{pmatrix}, \\ f(p_i^h) = 0. \end{cases} \quad (33)$$

$$\begin{cases} -6\eta L_h f(x) = g, \\ f(p_i^h) = 0. \end{cases} \quad (34)$$

where  $L_h$  is the seven-point Laplacian stencil with suitable boundary conditions and  $p_i^h$  is the landmark points on the grid  $V_h$ . To simplify, let  $\mathcal{L}_h = \begin{pmatrix} I & L_h \\ L_h & 6\frac{\eta}{\sigma}L_h \end{pmatrix}$  if  $\sigma \neq 0$  and  $\mathcal{L}_h = 6\eta L_h$  if  $\sigma = 0$ .

The boundary conditions depends on the setting of the original problem (16). Either Dirichlet, Neumann or the combination of both can be enforced. For example, suppose the unit cube is mapped to a unit cube, the boundary conditions can be set as follows. Let  $f = (f_1, f_2, f_3)$ . Then:

$$\begin{aligned}
f_1(0, y, z) &= 0 \text{ and } f_1(1, y, z) = 1, \\
f_2(x, 0, z) &= 0 \text{ and } f_2(x, 1, z) = 1, \\
f_3(x, y, 0) &= 0 \text{ and } f_3(x, y, 1) = 1.
\end{aligned} \tag{35}$$

Therefore, for  $f_1$ , we impose the Dirichlet boundary condition on  $\{0, 1\} \times [0, 1] \times [0, 1]$  and Neumann boundary condition on other boundaries. For  $f_2$  and  $f_3$ , we do it similarly.

The next question is how we set the landmark constraints on coarser levels. In other words, we need to choose vertices  $\{p_i\}_{i=1}^m$  such that  $f(p_i) = 0$ . On the finest grid, these points are chosen to be the original landmark points from the input. On the next coarser grid, these points are chosen to be the set of points belonging to the neighbourhood of landmark points at the previous finer level. For example, if we have a landmark point  $(0.375, 0.375, 0.375)$  on  $V_{2^{-2}}$ , then on  $V_{2^{-1}}$ , the set of points in the neighbourhood of it are  $(0.25, 0.25, 0.25)$ ,  $(0.25, 0.25, 0.5)$ ,  $(0.25, 0.5, 0.25)$ ,  $(0.25, 0.5, 0.5)$  and so on. All these points will be selected as landmark points on  $V_{2^{-1}}$ . Although this scheme would probably make the coarsest level having many landmark points, it does not cause either convergence or complexity problem because more landmark points means less free variables and faster convergence.

---

**Algorithm 2:**  $f = \text{Vcycle}_h(f, g)$

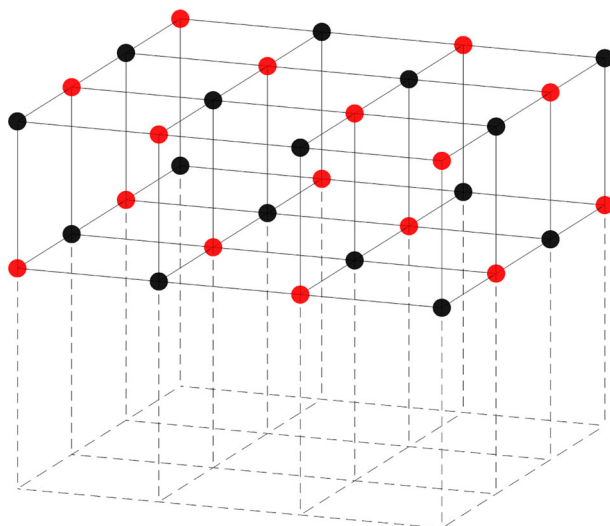
---

- 1 If  $h = 2^{-k}$  for some  $k \geq 1$ , return  $\mathcal{L}_h^{-1}g$ ;
  - 2  $f \leftarrow S(f, g)$ ;
  - 3  $f \leftarrow f + I_{2h}^h \text{Vcycle}_{2h}(0, I_{2h}^{2h}(g - \mathcal{L}_h f))$ ;
  - 4  $f \leftarrow S(f, g)$ ;
- 

The multi-grid V-cycle can now be described as follows. At level  $h$  (the grid with spacing  $h$ ), (33) and (34) can both be regarded as a linear system of the form  $\mathcal{L}_h f = \mathbf{c}$ . We first relax  $\mathcal{L}_h f = \mathbf{c}$  using certain iterative scheme, such as Jacobi, Gauss-Seidel or Successive over-relaxation methods. We denote the approximated solution  $\tilde{f}_h$  after the relaxation by  $\tilde{f}_h = S(f, \mathbf{c})$ . We then compute the residual  $r_h = \mathbf{c} - \mathcal{L}_h \tilde{f}_h$ . To improve the solution, we relax  $\mathcal{L}_{2h} e = r_{2h}$  on a coarser grid  $V_{2h}$ , where  $r_{2h} = I_{2h}^{2h}(r_h)$  is the projection of  $r_h$  from level  $h$  to level  $2h$  and  $I_{2h}^{2h}$  is the linear projection operator. Denote the approximated solution of  $\mathcal{L}_{2h} e = r_{2h}$  by  $e_{2h}$ . Then, the approximated solution of  $\mathcal{L}_h f = \mathbf{c}$  can be improved by  $\tilde{f}_h$  by  $\tilde{f}_h \leftarrow \tilde{f}_h + I_{2h}^h e_{2h}$ . This completes a V-cycle at level  $h$ . Note that when computing the approximated solution of  $\mathcal{L}_{2h} e = r_{2h}$ , we can again apply a multi-grid V-cycle on level  $2h$ . The detailed multi-grid V-cycle algorithm can now be described as in Algorithm 3.

The relaxation  $S$  in Algorithm 3 removes the high frequency component in the residual for  $r_h = \mathbf{c} - \mathcal{L}_h \tilde{f}_h$ . In this paper, the relaxation  $S$  is chosen to be the Red-black Gauss-Seidel (RBGS) iterations. We will briefly describe the RBGS iteration. For details, we refer the readers to [48]. The red-black Gauss-Seidel modifies the standard Gauss-Seidel method by reordering different equations. The basic idea is to group the grid points into two groups, identified as black and red nodes, such that black nodes are surrounded by red nodes only and vice versa. The red-black grouping of grid points in 3-dimensional space is as shown in Fig. 3. The Laplace operator  $\mathcal{L}_h$  under the red-black ordering of grid points can be rewritten as:  $\mathcal{L}_h = \begin{pmatrix} \mathbf{D}_r & \mathbf{U} \\ \mathbf{L} & \mathbf{D}_b \end{pmatrix}$ , where  $\mathbf{D}_r$  and  $\mathbf{D}_b$  are diagonal matrices associated to the red nodes and black nodes respectively. The Gauss-Seidel iteration can now be written as:

$$\begin{aligned}
\tilde{f}_h^{r,n+1} &= \mathbf{D}_r^{-1}(-\mathbf{U}\tilde{f}_h^{b,n} + \mathbf{c}_r), \\
\tilde{f}_h^{b,n+1} &= \mathbf{D}_b^{-1}(-\mathbf{L}\tilde{f}_h^{r,n+1} + \mathbf{c}_b)
\end{aligned} \tag{36}$$



**Fig. 3** Illustration of *red-black* ordering of grid points for 3-dimensional space (Color figure online)

where  $\tilde{f}_h^{r,n}$  and  $\tilde{f}_h^{b,n}$  are the components of  $\tilde{f}_h$  associated to the red and black nodes respectively at the  $n$ -th iterations.  $\mathbf{c}_r$  and  $\mathbf{c}_b$  are the components of  $\tilde{c}$  associated to the red and black nodes respectively. As a result, instead of solving a triangular system as in the standard Gauss-Seidel iterations, we perform matrix-vector products and vector scaling operations with half as many variables in each iteration.

In our implementation, we have chosen  $S$  to be four iterations of Red-Black Gauss-Seidel method. The red-black ordering for step 2 in Algorithm 3 is the opposite to the red-black ordering for step 4. This reverse ordering ensure the obtained multi-grid V-cycle preconditioner  $M$  to be symmetric positive definite for applying PCG [50].

For the restriction and interpolation operator, the full weighting restriction and bilinear interpolation operator are used. Note that the result of the interpolation operator satisfies the landmark points condition because of our choices of landmark points on the coarse grid.

The overall  $\text{Vcycle}_h(0, g)$  is a linear operator on  $\tilde{f}$ . We simply write it as  $M\tilde{f}$ .  $M$  is our desired preconditioner. With  $M$ , we apply the PCG method with the preconditioner matrix  $M$  to solve the original f-subproblem (28) [53]. This can be described as follows.

---

**Algorithm 3:** *Solving (28)*

---

- 1 Denote  $Mg = \text{Vcycle}_h(0, g)$ ;
  - 2 Apply the preconditioned conjugate gradient method on (28) with the preconditioner matrix  $M$ ;
- 

## 4.2 R-subproblem

The R-subproblem in Algorithm 1 is a tetrahedron-wise problem. Therefore, parallel computing can be adopted in this subproblem. More explicitly, we want to find  $R(T)$  on each tetrahedron  $T$  which minimizes the following energy

$$\min_{R(T) \in \mathbb{R}^{3 \times 3}, \det(R(T)) > 0} \left( \frac{\|Df^k\|_F^2}{\det(R(T))^{2/3}} + \frac{\eta}{2} \|R(T) - B\|_F^2 \right) \quad (37)$$

where  $B = Df^k - \lambda^k$ .

Let the SVD of  $B$  be  $U \Sigma V^*$ . In the case when  $R$  can be written as  $U \tilde{\Sigma} V^*$ , where  $\tilde{\Sigma}$  is an unknown diagonal matrix. Problem (37) can be much simplified. Since we want  $\det(R(T)) > 0$ ,  $\tilde{\Sigma}$  must satisfy  $\text{sgn}(\det(\tilde{\Sigma})) = \text{sgn}(\det(UV^*))$ . Denote  $\Gamma = \{\tilde{\Sigma} \mid \text{sgn}(\det(\tilde{\Sigma})) = \text{sgn}(\det(UV^*))\}$ . By substituting these representations and constraint to problem (37), we have the following optimization problem with three variables:

$$\min_{\tilde{\Sigma} \in \Gamma} \left( \frac{\|Df^k\|_F^2}{\det(\tilde{\Sigma})^{2/3}} + \frac{\eta}{2} \|\tilde{\Sigma} - \Sigma\|_F^2 \right) \quad (38)$$

In other words, by making the assumption that  $R$  can be written as  $U \tilde{\Sigma} V^*$ , we can simply the original R-subproblem as (38). A natural question is the relationship between the problem (38) and our original R-subproblem (37). The following theorem gives the answer.

**Theorem 1** Suppose  $\bar{\Sigma}$  is the minimizer of (38). Then,  $R = U \bar{\Sigma} V^*$  is the minimizer of (37), where  $B = U \Sigma V^*$  is the SVD of  $B$ .

*Proof* This is related to the general two-sided Procrustes problem. Suppose  $X_1$  and  $X_2$  are  $n \times n$  matrices. Define:

$$E_P(Q_1, Q_2) = \|Q_1^* X_1 Q_2 - X_2\|_F^2 \quad (39)$$

where  $Q_1$  and  $Q_2$  are  $n \times n$  orthogonal matrices. Let  $X_1 = P_1 \Sigma_1 R_1^*$  and  $X_2 = P_2 \Sigma_2 R_2^*$  be the SVDs of  $X_1$  and  $X_2$  respectively. Then, the minimizer of  $E_P$  satisfies:

$$P_1 = Q_1 P_2 \Pi; \quad R_1 = Q_2 R_2 \Pi, \quad (40)$$

where  $\Pi$  is the permutation matrix that maximizes  $\text{Tr}(\Sigma_2^* \Pi^* \Sigma_1 \Pi)$  (see p. 89–90 in [20]).

Let  $B = U \Sigma V^*$  be the SVD of  $B$  and let  $P \tilde{\Sigma} Q$  be the SVD of  $U^* R V$ . Our R-subproblem (37) is equivalent to minimizing:

$$E_R^{new}(\tilde{P}, \tilde{\Sigma}, \tilde{Q}) := \left\{ \|\tilde{P} \tilde{\Sigma} \tilde{Q}^* - \Sigma\|_F^2 + \frac{c}{\det(\tilde{\Sigma})^{2/3}} \right\}, \quad (41)$$

for some positive constant  $c$ .

Let  $\bar{\Sigma}$  be the minimizer of:

$$\min_{\bar{\Sigma}} \left\{ \|\bar{\Sigma} - \Sigma\|_F^2 + \frac{c}{\det(\bar{\Sigma})^{2/3}} \right\}. \quad (42)$$

Fixing a diagonal matrix  $D$ , we consider the minimization problem over  $(\tilde{P}, \tilde{Q})$  of  $E_R^{new}(\tilde{P}, D, \tilde{Q})$ . According to (40), the minimizer must satisfy  $I = \tilde{P} \Pi$  and  $I = \tilde{Q} \Pi$ . Thus, for any orthogonal matrices  $P$  and  $Q$  and diagonal matrix  $D$ ,

$$\begin{aligned} E_P^{new}(P, D, Q) &= \|P D Q^* - \Sigma\|_F^2 + \frac{c}{(\det(D))^{2/3}} \\ &\geq \|\Pi^* D \Pi - \Sigma\|_F^2 + \frac{c}{(\det(\Pi^* D \Pi))^{2/3}} \\ &\geq \|\bar{\Sigma} - \Sigma\|_F^2 + \frac{c}{(\det(\bar{\Sigma}))^{2/3}} \\ &= E_R^{new}(I, \bar{\Sigma}, I). \end{aligned} \quad (43)$$



Thus,  $(I, \bar{\Sigma}, I)$  is a minimizer of (41). We conclude that:  $R = UI\bar{\Sigma}IV^* = U\bar{\Sigma}V^*$  is a minimizer of (37).  $\square$

**Theorem 2** *The Euler Lagrange equation of (38) is:*

$$\tilde{\Sigma} - \frac{a}{(\det(\tilde{\Sigma}))^{2/3}} \tilde{\Sigma}^{-1} = \Sigma, \quad \text{where } a = \frac{2\|Df^k\|^2}{3\eta}. \quad (44)$$

*Proof* Let  $x_i$  be the diagonal of  $\Sigma$  and  $y_i$  be the diagonal of  $\tilde{\Sigma}$ . Denote  $\tilde{y}_1 = y_1 + \epsilon\lambda_1$  to be the variation of  $y_1$ . Consider the derivative of energy with respect to  $\epsilon$ , we have

$$\begin{aligned} \frac{d}{d\epsilon} \frac{\|Df^k\|^2}{(\tilde{y}_1 y_2 y_3)^{2/3}} + \frac{\eta}{2} \left[ (\tilde{y}_1 - x_1)^2 + (y_2 - x_2)^2 + (y_3 - x_3)^2 \right] \Big|_{\epsilon=0} &= 0 \\ \Rightarrow \lambda_1 \left( \frac{-a}{(y_1 y_2 y_3)^{2/3}} \left( \frac{1}{y_1} \right) + y_1 - x_1 \right) &= 0. \end{aligned} \quad (45)$$

Since  $\lambda_1$  is arbitrary, we have

$$\frac{-a}{(y_1 y_2 y_3)^{2/3}} \left( \frac{1}{y_1} \right) + y_1 - x_1 = 0 \quad (46)$$

Similar equations can be obtained for the variations of  $y_2$  and  $y_3$ . By combining the results, we have the same formula as in (44).  $\square$

To tackle with the nonlinear recurrence Eq. (44), we propose Algorithm 4 below that gives the solution of (47) to obtain a minimizer of the optimization problem (38). More specifically, since the system (44) is coupled by the term  $\det(\tilde{\Sigma})$ , we can solve the equation iteratively by

$$\tilde{\Sigma}_n - \frac{a}{(\det(\tilde{\Sigma}_{n-1}))^{2/3}} \tilde{\Sigma}_n^{-1} = \Sigma \quad (47)$$

where  $\tilde{\Sigma}_n$  is the  $\tilde{\Sigma}$  in step  $n$ . Define  $D_n = \det(\tilde{\Sigma}_{n-1})^{2/3}$ . By element-wise decoupling the nonlinear recurrence Eq. (47), we have the quadratic equations  $y_i^n - \frac{a}{D} (y_i^n)^{-1} = x_i$  for  $i = 1, 2, 3$ , where  $\tilde{\Sigma}_n = \text{diag}(y_1^n, y_2^n, y_3^n)$ . Solving the quadratic equations, we have

$$y_i^n(D_n) = \frac{x_i \pm \sqrt{x_i^2 + \frac{4a}{D_n}}}{2}, \quad i = 1, 2, 3 \quad (48)$$

where the sign is chosen according to Algorithm 4. The motivation and the convergence analysis of the proposed iteration scheme is explained in Theorem 3.

Before introducing Theorem 3, the following lemma is necessary.

**Lemma 1** *Let  $\sigma \in \mathbb{R}_+^n$  be a vector with positive values. The function*

$$f(\sigma) = \frac{c}{\prod_{i=1}^n \sigma_i^{2/\eta}} + \eta \sum_{i=1}^n (\sigma_i - a_i)^2, \quad c > 0 \text{ and } \eta > 0 \quad (49)$$

*is convex.*

*Proof* Recall that the log barrier function  $B(u) = -\log(\det(u))$ , where  $u$  is a symmetric positive definite matrix, is convex [3]. This implies that  $\frac{2}{\eta} B(D)$ , where  $D$  is a diagonal matrix with positive diagonal elements  $\{d_{ii}\}_{i=1}^n$ , is also a convex in  $D$ . Note that

**Algorithm 4:** *Solving (47)*


---

```

1 Compute the SVD of  $B = U \Sigma V^*$  where the diagonal of  $\Sigma$  is  $x_i \geq 0$ ;
2 Set  $D^1 = (\det(R^{(\text{last})})^{2/3})$ . Denote  $D_{n+1} = (y_1^n y_2^n y_3^n)^{2/3}$ ;
3 repeat
4   Set  $y_i^n = \frac{1}{2} \left( x_i + \sqrt{x_i^2 + \frac{4a}{D_n}} \right)$  for  $i = 1, 2, 3$ ;
5   If  $\det(UV^*) < 0$ , set  $y_i^n = \frac{1}{2} \left( x_i - \sqrt{x_i^2 + \frac{4a}{D_n}} \right)$  for  $i = \text{argmin}_i x_i$ ;
6    $D^{n+1} \leftarrow \frac{1}{2} \left( D_n + (y_1^n y_2^n y_3^n)^{2/3} \right)$ ;
7    $n \leftarrow n + 1$ ;
8 until  $\|D_{n+1} - D_n\|_\infty < \varepsilon$ ;
9  $R = U \tilde{\Sigma} V^*$  where the diagonal entries of  $\tilde{\Sigma}$  are  $y_i$ ;

```

---

$$\exp\left(-\frac{2}{n}B(D)\right) = \exp\left(-\frac{2}{n}\log(\det(u))\right) = \exp\left(\log\left(\prod_{i=1}^n \frac{1}{d_{ii}^{2/n}}\right)\right) = \frac{1}{\prod_{i=1}^n d_{ii}^{2/n}}. \quad (50)$$

As the exponential of a convex function is also convex and  $c > 0$ , we have shown that  $\frac{c}{\prod_{i=1}^n \sigma_i^{2/n}}$  is convex.

Define  $g(\sigma) = \sum_{i=1}^n (\sigma_i - a_i)^2$ . We have

$$\frac{\partial^2 g}{\partial \sigma_j \partial \sigma_i} = \begin{cases} 2 & \text{if } i = j, \\ 0 & \text{otherwise} \end{cases} \quad (51)$$

Therefore, the Hessian matrix of  $g$  is equal to  $2I_n$ , where  $I_n$  is the  $n$ -dimensional identity matrix. Therefore  $g$  is also convex in  $\sigma$ . By combining both results, we can conclude that  $f(\sigma)$  is a convex function.  $\square$

The above lemma states that the simplified optimization problem (38) is convex in the positive octant region. In fact, using the same argument, we can show that the optimization problem (38) is convex in any one of the octant regions. Hence, the optimization is a global minimizer in each octant region.

Now, we will explain the convergence of Algorithm 4 to the minimizer of the optimization problem (38).

**Theorem 3** *Given any  $3 \times 3$  matrix  $B$ ,  $a > 0$ . Algorithm 4 converges linearly to a solution of the nonlinear recurrence Eq. (44) with the rate  $\frac{1}{2}$ , which is a minimizer of (38).*

*Proof* Recall that  $y_i^n$  at the  $n^{\text{th}}$  iteration is defined as follows:

$$y_i^k(D_n) = \frac{x_i \pm \sqrt{x_i^2 + \frac{4a}{D_n}}}{2}. \quad (52)$$

The sign of  $y_i^n$  is chosen as to minimize the energy functional (38), which is given by the following:

$$\begin{aligned} \min_{\tilde{\Sigma} \in \Gamma} \left( \frac{\|Df^k\|^2}{\det(\tilde{\Sigma})^{2/3}} + \frac{\eta}{2} \|\tilde{\Sigma} - \Sigma\|_F^2 \right) &= \eta \min_{\tilde{\Sigma} \in \Gamma} \left( \frac{a}{\frac{2}{3} \det(\tilde{\Sigma})^{2/3}} + \frac{1}{2} \|\tilde{\Sigma} - \Sigma\|_F^2 \right) \\ &= \eta \min_{\tilde{\Sigma} \in \Gamma} \left( \frac{3a}{2(y_1 y_2 y_3)^{2/3}} + \frac{1}{2} \sum (x_i - y_i)^2 \right). \end{aligned} \quad (53)$$

Our goal is to make  $D = (y_1 y_2 y_3)^{2/3}$  larger and  $y_i$  closer to  $x_i$ . Therefore, the sign appears in Eq. (48) can be determined according to the magnitude of the energy. If  $\det(UV^*) > 0$ , we can either set  $+$  sign in Eq. (48) for all  $i$  or we set  $-$  sign for only two of  $y_i$ . However, we can eliminate the second case by the following argument. Note that the second term in the energy functional dominates the overall energy and  $x_i \geq 0$  for all  $i$ , we have

$$\left(x_i - \frac{x_i + \sqrt{x_i^2 + \frac{4a}{D}}}{2}\right)^2 \leq \left(x_i - \frac{x_i - \sqrt{x_i^2 + \frac{4a}{D}}}{2}\right)^2 \quad \forall i. \quad (54)$$

Therefore, the minimizer should satisfy the  $+$  sign in Eq. (48) for all  $i$ . If  $\det(UV^*) < 0$ , we can either set  $-$  sign in Eq. (48) for all  $i$  or we set  $-$  sign for one of the  $y_i$ . Similar argument can be made and the minimizer should satisfy the  $+$  sign in Eq. (48) for all  $i \neq \operatorname{argmin}_i x_i$ . This explains the purpose of step 5. Without loss of generality, we assume  $\operatorname{argmin}_i x_i = 1$ .

Let  $F(D) = (y_1(D)y_2(D)y_3(D))^{2/3}$  and  $G(D) = \frac{D+F(D)}{2}$ . We have

$$F'(D) = \frac{-2a}{3D^2} F(D) \sum_{i=1}^3 \frac{\operatorname{sgn}(y_i)}{y_i} \frac{1}{\sqrt{x_i^2 + \frac{4a}{D}}} \quad (55)$$

Note that each term  $\frac{\operatorname{sgn}(y_i)}{y_i} \frac{1}{\sqrt{x_i^2 + \frac{4a}{D}}}$  in the sum is positive. Hence  $F'(D) < 0$ .

For the case  $\det(UV^*) > 0$ , we have

$$-F'(D) = \frac{2a}{3D^2} F(D) \sum_{i=1}^3 \frac{1}{y_i} \frac{1}{\sqrt{x_i^2 + \frac{4a}{D}}} < \frac{2a}{3D^2} F(D) \sum_{i=1}^3 \frac{D}{2a} = \frac{F(D)}{D}. \quad (56)$$

For the case  $\det(UV^*) < 0$ , we have

$$\frac{\operatorname{sgn}(y_1)}{y_1 \sqrt{x_1^2 + \frac{4a}{D}}} = \frac{2}{\sqrt{x_1^2 + \frac{4a}{D}} - x_1} \frac{1}{\sqrt{x_1^2 + \frac{4a}{D}}} = \frac{D}{2a} \frac{x_1 + \sqrt{x_1^2 + \frac{4a}{D}}}{\sqrt{x_1^2 + \frac{4a}{D}}} \quad (57)$$

when  $i = 1$ . For  $i = 2, 3$ , we have

$$\frac{1}{y_i \sqrt{x_i^2 + \frac{4a}{D}}} = \frac{2}{x_i^2 + \frac{4a}{D}} \frac{\sqrt{x_i^2 + \frac{4a}{D}}}{x_i + \sqrt{x_i^2 + \frac{4a}{D}}} \leq \frac{D}{2a} \frac{\sqrt{x_1^2 + \frac{4a}{D}}}{x_1 + \sqrt{x_1^2 + \frac{4a}{D}}}. \quad (58)$$

Hence we have

$$-F'(D) \leq \frac{2a}{3D^2} F(D) \frac{D}{2a} \left( \frac{x_1 + \sqrt{x_1^2 + \frac{4a}{D}}}{\sqrt{x_1^2 + \frac{4a}{D}}} + 2 \frac{\sqrt{x_1^2 + \frac{4a}{D}}}{x_1 + \sqrt{x_1^2 + \frac{4a}{D}}} \right). \quad (59)$$

Since  $\frac{1}{2} < \frac{\sqrt{x_1^2 + \frac{4a}{D}}}{x_1 + \sqrt{x_1^2 + \frac{4a}{D}}} < 1$ , by considering the function  $\frac{1}{x} + 2x$  on the interval  $(\frac{1}{2}, 1)$ , we conclude that the last term inside the parenthesis is less than 3. Hence, in both cases, we have  $-F'(D) \leq \frac{F(D)}{D}$ .

Now, let  $\tilde{D}$  be the solution of  $G(D) = D$ . We proceed to show that  $\{D_n\}_{n=1}^\infty$  converges to  $\tilde{D}$ .

Consider the case when  $D_n \geq \tilde{D}$ . Since  $G'(D) = \frac{1}{2} + \frac{F'(D)}{2}$ , we have

$$\frac{1}{2} \geq G'(D) \geq \frac{1}{2} - \frac{F(D)}{2D} \geq \frac{1}{2} - \frac{F(\tilde{D})}{2D} = \frac{1}{2} - \frac{\tilde{D}}{2D} \geq 0. \quad (60)$$

This suggests  $G$  is an increasing function in  $D$ . Also,  $\tilde{D} = G(\tilde{D}) = \frac{\tilde{D}+F(\tilde{D})}{2}$  implies  $\tilde{D} = F(\tilde{D})$ . Hence,  $\frac{D+F(\tilde{D})}{2} = \frac{D+\tilde{D}}{2} \geq \frac{D+F(D)}{2} = G(D) \geq \tilde{D}$  for  $D \geq \tilde{D}$ . We get that

$$D_n \geq \frac{D_n + \tilde{D}}{2} \geq G(D_n) = D_{n+1} \geq \tilde{D}. \quad (61)$$

$\{D_n\}_{n=1}^\infty$  is thus a decreasing sequence converging to some  $D^* \geq \tilde{D}$ . Also, from the previous inequalities, we observe that  $\frac{D^*+\tilde{D}}{2} \geq D^* \geq \tilde{D}$ , which gives  $\tilde{D} \geq D^* \geq \tilde{D}$ . We conclude that  $D^* = \tilde{D}$ .

For the case  $D_n \leq \tilde{D}$ , we have  $G(D) = \frac{D+F(D)}{2} \geq \frac{D+\tilde{D}}{2}$ . Thus,

$$G(D_n) = D_{n+1} \geq \frac{D_n + \tilde{D}}{2} \geq D_n. \quad (62)$$

Suppose  $D_k \leq \tilde{D}$  for all  $k > n$ . Using a similar argument as before, we conclude that  $\{D_n\}_{n=1}^\infty$  is an increasing sequence converging to  $\tilde{D}$ . Suppose  $D_k > \tilde{D}$  for some  $k > n$ . From the previous conclusion, we can also get the same convergence result. That is,  $D_n \rightarrow \tilde{D}$ .

In both cases, we can show that  $\frac{|D_{n+1}-D_n|}{|D_n-D_{n-1}|} \leq \frac{1}{2}$ . As a result, the sequence  $\{D_n\}_{n=1}^\infty$  converges at a rate  $\frac{1}{2}$ . Also, since  $\tilde{\Sigma}_n = \text{diag}(y_1^n(D_n), y_2^n(D_n), y_3^n(D_n))$  depends on  $D_n$ ,  $\tilde{\Sigma}_n$  converges to a solution of the nonlinear recurrence Eq. (44) with a rate  $\frac{1}{2}$ .

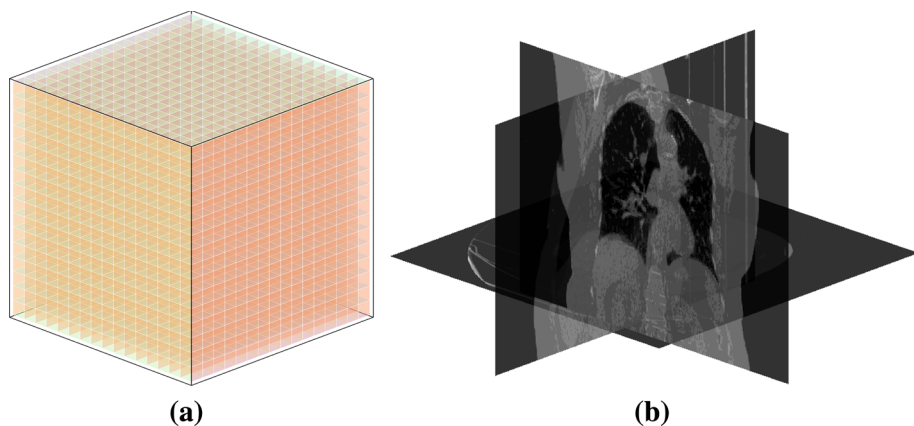
In other words, algorithm 4 converges to a solution of the Euler-Lagrange equation of (38). Depending on the sign of  $\det(UV^*)$ , our algorithm search for a critical point at the first octant region if  $\det(UV^*) > 0$  and at the other octant (with  $x_i < 0$ ) if  $\det(UV^*) < 0$ . According to Lemma 1, the critical point must be the global minimizer of (38) in the corresponding octant region. Hence, algorithm 4 converges to the minimizer of (38).  $\square$

## 5 Experimental Result

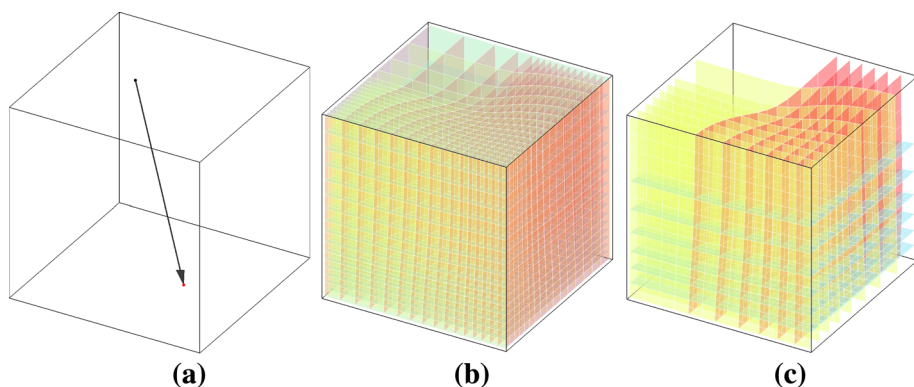
To validate the effectiveness of our proposed algorithm, experiments on synthetic examples have been carried out to compute 3-dimensional quasi-conformal landmark-matching transformation. We have also applied our proposed algorithm on lung CT images with respiratory deformations. Experimental results are reported in this section.

### 5.1 Synthetic Examples

We first test our algorithm to compute the landmark-matching transformation with one landmark. Figure 5a shows how the landmark point is deformed. The deformation of the landmark point is large. The point  $p_1 = [0.6, 0.6, 0.6]$  is moved to  $q_1 = f(p_1) = [0.3, 0.3, 0.3]$ . Using the proposed algorithm, we obtain a diffeomorphic transformation that satisfies the landmark constraint exactly. Figure 5b shows the obtained transformation. It is visualized by the deformation of the original reference mesh as shown in Fig. 4a under the obtained transformation. The reference mesh is a regular grid of a cube discretizing the source domain. Figure 5c shows the visualization of the obtained transformation with a sparser view (to better demonstrate the transformation). Note that we set  $\Omega_1 = \Omega_2 = \Omega = [0, 1]^3$  in all our synthetic



**Fig. 4** The regular reference mesh of the cube and the 3 dimensional lung CT image. **a** Regular mesh on a cube, **b** CT image of a lung



**Fig. 5** One-point landmark-matching experiment. **a** Landmark, **b** QC deformation, **c** sparse view

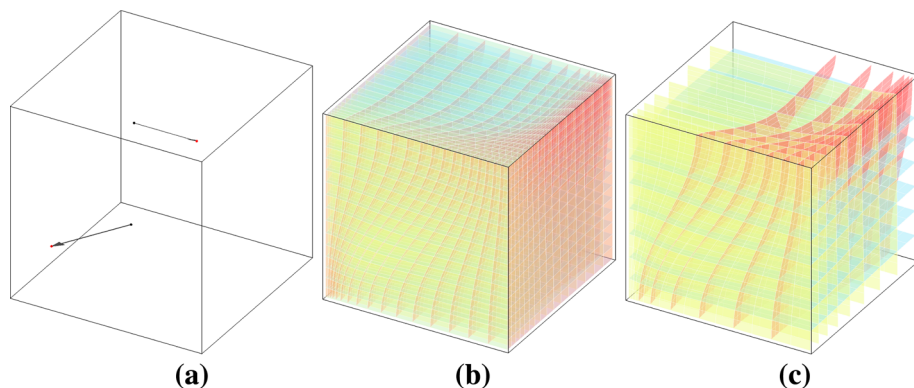
experiments. By the boundary setting as discussed in Sect. 4.1, the image of the resultant map is restricted to be the cube  $\Omega$ , even though the landmark moves towards the boundary.

Secondly, we test the algorithm to compute the landmark-matching transformation with two landmarks moving towards different directions. Deformations of both landmark points are large, as shown in Fig. 6a. More specifically, two points  $p_i$  are moved to  $q_i = f(p_i)$  as follows:

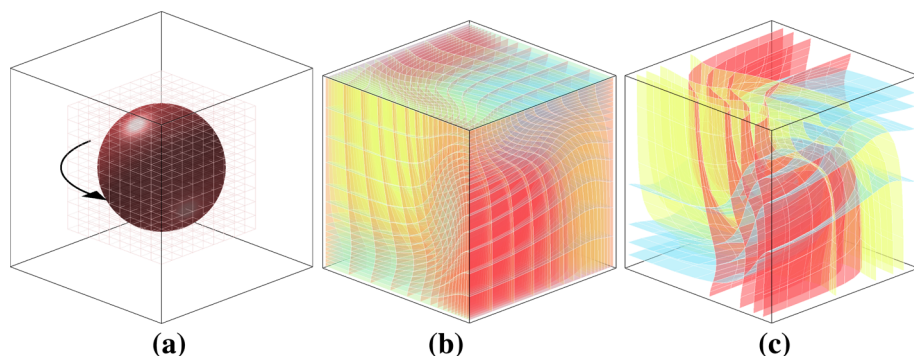
$$\begin{bmatrix} p_1(x) & p_1(y) & p_1(z) \\ p_2(x) & p_2(y) & p_2(z) \end{bmatrix} = \begin{bmatrix} 0.6 & 0.7 & 0.7 \\ 0.4 & 0.6 & 0.3 \end{bmatrix} \rightarrow \begin{bmatrix} 0.3 & 0.2 & 0.9 \\ 0.2 & 0.9 & 0.2 \end{bmatrix} = \begin{bmatrix} q_1(x) & q_1(y) & q_1(z) \\ q_2(x) & q_2(y) & q_2(z) \end{bmatrix} \quad (63)$$

Figure 6b shows the obtained transformation. Figure 6c shows the visualization of the obtained transformation with a sparser view.

We also test the algorithm to compute the landmark-matching transformation with an inner ball being chosen as landmarks. Points  $p_i = (p_i(x), p_i(y), p_i(z))$  inside the inner ball are moved by the following transformation:



**Fig. 6** Two-point landmark-matching experiment. **a** Landmark, **b** QC deformation, **c** sparse view



**Fig. 7** Landmark-matching experiment to register *rotating ball*. **a** Landmark, **b** QC deformation, **c** sparse view

$$\begin{bmatrix} q_i(x) \\ q_i(y) \\ q_i(z) \end{bmatrix} = \begin{bmatrix} f_1(p_i) \\ f_2(p_i) \\ f_3(p_i) \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_i(x) - 0.5 \\ p_i(y) - 0.5 \\ p_i(z) - 0.5 \end{bmatrix} \quad (64)$$

In other words, points inside the sphere are chosen as landmarks and they are rotated anti-clockwisely, as shown in Fig. 7a. The obtained landmark-matching transformation, which is visualized as the deformation of the standard grid by the transformation, is shown in Fig. 7b. Figure 7c visualizes the obtained transformation with a sparser view. Note that the obtained transformation is folding-free (Please refer to Table 1 which will be described later).

Next, we test the algorithm on an example of which all the points on a plane

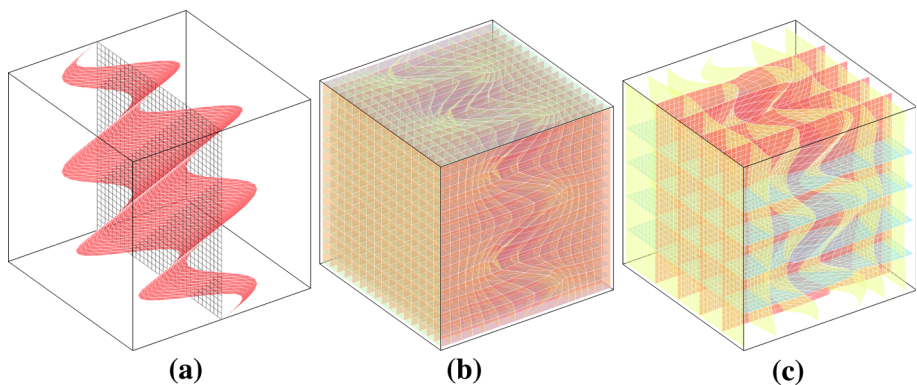
$$p_i = [0.5, p_i(y), p_i(z)] \quad \forall (p_i(y), p_i(z)) \in [0, 1] \times [0, 1]$$

are chosen as landmarks (grey plane in Fig. 8a). The landmarks are deformed to a wave-shape surface (red surface in Fig. 8b) by the following transformation:

$$\begin{bmatrix} q_i(x) \\ q_i(y) \\ q_i(z) \end{bmatrix} = \begin{bmatrix} f_1(p_i) \\ f_2(p_i) \\ f_3(p_i) \end{bmatrix} = \begin{bmatrix} 0.5 + \frac{1}{5} \sin(4\pi(p_i(y) + p_i(z))) \\ p_i(y) \\ p_i(z) \end{bmatrix}. \quad (65)$$

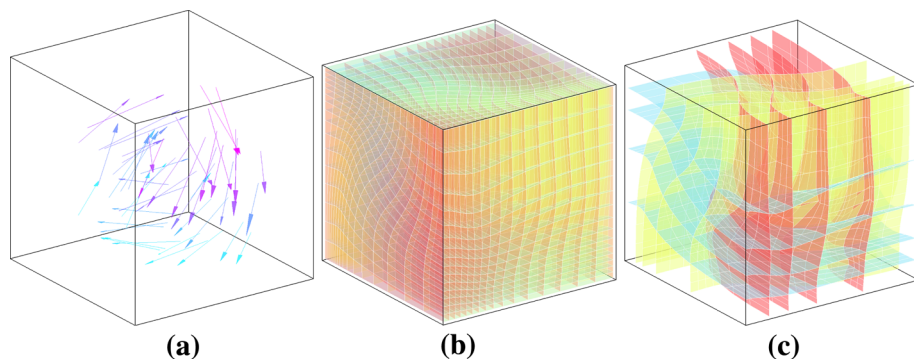
**Table 1** Quantitative measures of the registration experiment

|            | LM <sub>max</sub><br>LM <sub>mean</sub> | Proposed     |                     |  | Thin plate spline |                     |  |
|------------|---|--------------|---------------------|--|-------------------|---------------------|--|
|            |   | max K<br>#LM | min Det<br>Time (s) | e <sub>max</sub> /e <sub>mean</sub><br>#Fold | max K<br>#LM      | min Det<br>Time (s) | e <sub>max</sub> /e <sub>mean</sub><br>#Fold |
| One-point  | 0.5196                                  | 8.9756       | 0.2519              | 0 / 0  | ∞                 | −0.0033             | 0.0017 / 0.0017                              |
|            | 0.5196                                  | 1            | 11.3644 s           | 0  | 1                 | 0.08727 s           | 2  |
| Two-point  | 0.6164                                  | 2.9295       | 0.0989              | 0 / 0  | ∞                 | −0.0722             | 0.0383 / 0.0352                              |
|            | 0.4953                                  | 2            | 11.7554 s           | 0  | 2                 | 0.0134 s            | 213  |
| Twist      | 0.5194                                  | 4.1488       | 0.1753              | 0 / 0  | ∞                 | −0.0848             | 0.0007 / 0.0004                              |
|            | 0.3383                                  | 50           | 18.2556 s           | 0  | 50                | 0.0810 s            | 131  |
| Rotate     | 0.5097                                  | 3.1939       | 0.4006              | 0 / 0  | ∞                 | −0.0755             | 0.0008 / 0.0004                              |
|            | 0.3071                                  | 3743         | 28.7540 s           | 0  | 3743              | 61.0223 s           | 35337  |
| Wave-shape | 0.2000                                  | 3.3007       | 0.3642              | 0 / 0  | ∞                 | −0.0682             | 0.0007 / 0.0005                              |
|            | 0.1256                                  | 1089         | 30.0948 s           | 0  | 1089              | 2.7450 s            | 4603   |
| CT1        | 0.0631                                  | 2.2204       | 0.1606              | 0 / 0  | 1.0640            | 0.0085              | 0.0121/0.0074                                |
|            | 0.0153                                  | 300          | 112.3901 s          | 0  | 300               | 1.3031 s            | 0  |
| CT2        | 0.0624                                  | 2.3331       | 0.1406              | 0 / 0  | 1.0740            | 0.0843              | 0.0136/0.0077                                |
|            | 0.0263                                  | 300          | 103.3852 s          | 0  | 300               | 1.1630 s            | 0  |
| CT3        | 0.0891                                  | 1.7137       | 0.3973              | 0 / 0  | 1.1763            | 0.0593              | 0.0123 / 0.0077                              |
|            | 0.0314                                  | 300          | 90.1452 s           | 0  | 300               | 1.1706 s            | 0  |
| CT4        | 0.0816                                  | 6.1340       | 0.0312              | 0 / 0  | 1.3528            | 0.0639              | 0.0138 / 0.0079                              |
|            | 0.0393                                  | 300          | 81.7555 s           | 0  | 300               | 1.2033 s            | 0  |
| CT5        | 0.0920                                  | 6.5297       | 0.0229              | 0 / 0  | 1.3226            | 0.0384              | 0.0159 / 0.0073                              |
|            | 0.0232                                  | 300          | 228.0194 s          | 0  | 300               | 1.5251 s            | 0  |

**Fig. 8** Landmark-matching experiment to register wave deformation. **a** Landmark, **b** QC deformation, **c** sparse view

Using our proposed algorithm, we obtain a transformation that satisfies the landmark constraints. Figure 8b shows the obtained transformation, which is bijective (Refer to Table 1). Figure 8c shows the transformation with a sparser view.





**Fig. 9** Landmark-matching experiment of random points with twisting deformation. **a** Landmark, **b** QC deformation, **c** sparse view

Finally, we test the algorithm to compute the landmark-matching transformation with random points being chosen as landmarks. These random landmark points are twisted by the following transformation (See Fig. 9a):

$$\begin{aligned} q_i(x) &= f_1(p_i) = p_i(x) - p_i(x) \cdot A(p_i(x), p_i(y)), \\ q_i(y) &= f_2(p_i) = \rho - p_i(z) \cdot A(\rho, p_i(z)), \\ q_i(z) &= f_3(p_i) = p_i(z) + q_i(y) \cdot A(\rho, p_i(z)). \end{aligned} \quad (66)$$

where

$$\begin{aligned} \rho &= p_i(y) + q_i(x) \cdot A(p_i(x), p_i(y)) \\ A(x, y) &= \frac{1}{100} \left( \frac{(\cos(\pi x) + 1)(\cos(\pi y) + 1)}{4} + \cos\left(\frac{\pi x}{2}\right) \cos\left(\frac{\pi y}{2}\right) \right) \end{aligned} \quad (67)$$

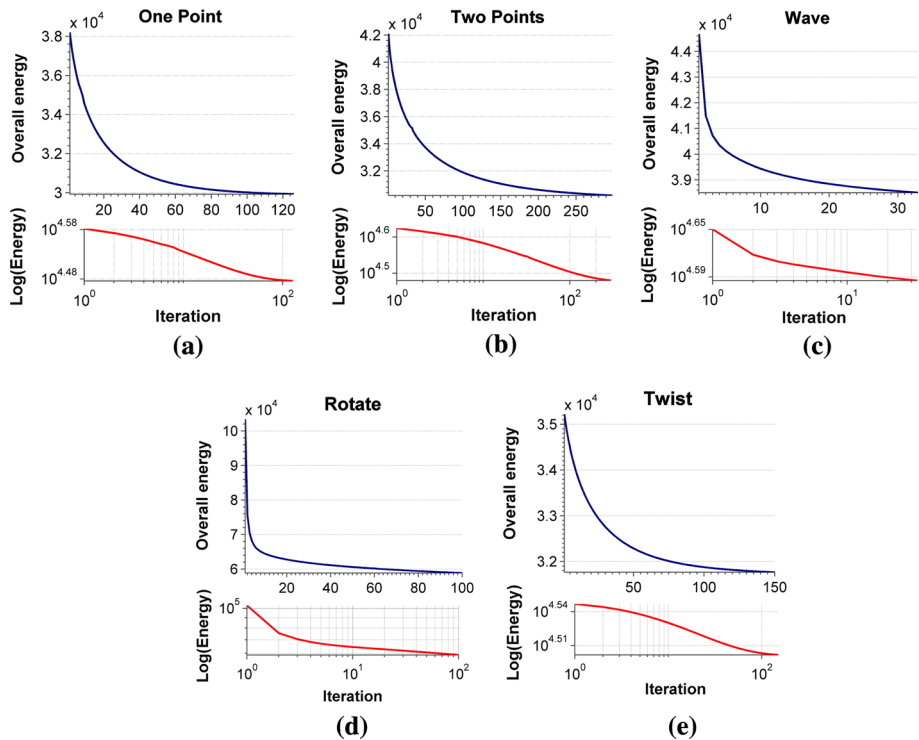
The twisting deformation is large and complicated. Using our algorithm, we are able to obtain a diffeomorphic landmark-matching transformation. Figure 9b shows the obtained transformation. Figure 9c shows the registration with a sparser view.

The upper row of Fig. 10a–e shows the overall energy (See 12 versus iterations for the “one point landmark”, “two-point landmark”, “wave-shape deformation”, “rotate sphere” and “twist point sets” examples respectively). Note that the overall energy of each mapping is iteratively reduced with a trend of converging to an optimal map with respect to our proposed model 12. The second row shows the corresponding log-log plot of the overall energy versus iterations. The negative slope appear in all five examples indicates that our proposed algorithm successfully minimizes the generalized conformality distortion  $K(f)$  while matching the prescribed landmark correspondences.

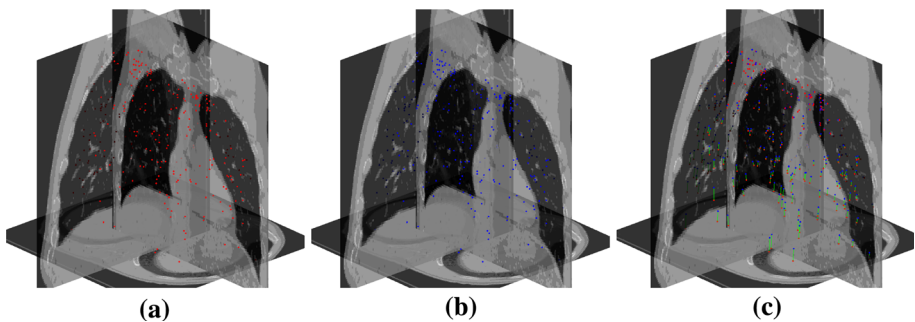
The above examples demonstrate that our proposed algorithm is effective for computing landmark-matching folding-free transformation with larger deformations. It works well even with large number of landmarks or large deformations.

## 5.2 Lung CT Landmark-Based Image Registration

We have also applied our algorithm to compute landmark-matching transformation of real four dimensional lung CT data with prescribed landmark correspondences at different times. Five sets of lung CT images are registered using our proposed algorithm. We choose the maximum inhalation phase image (at time  $t = 00$ ) and the maximum exhalation phase

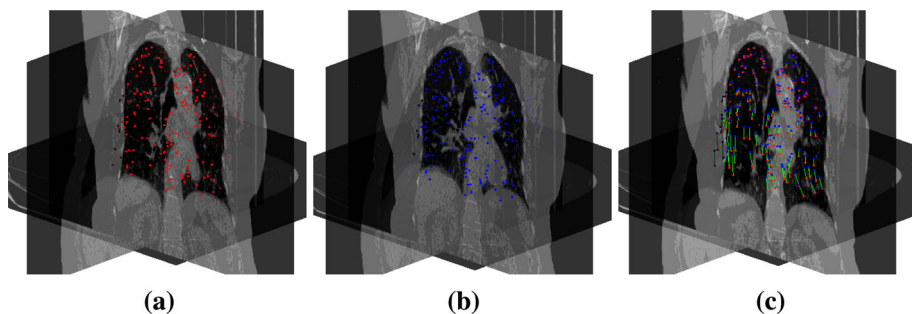


**Fig. 10** The overall energy versus iterations. **a** One-point, **b** two-point, **c** wave, **d** rotate, **e** twist

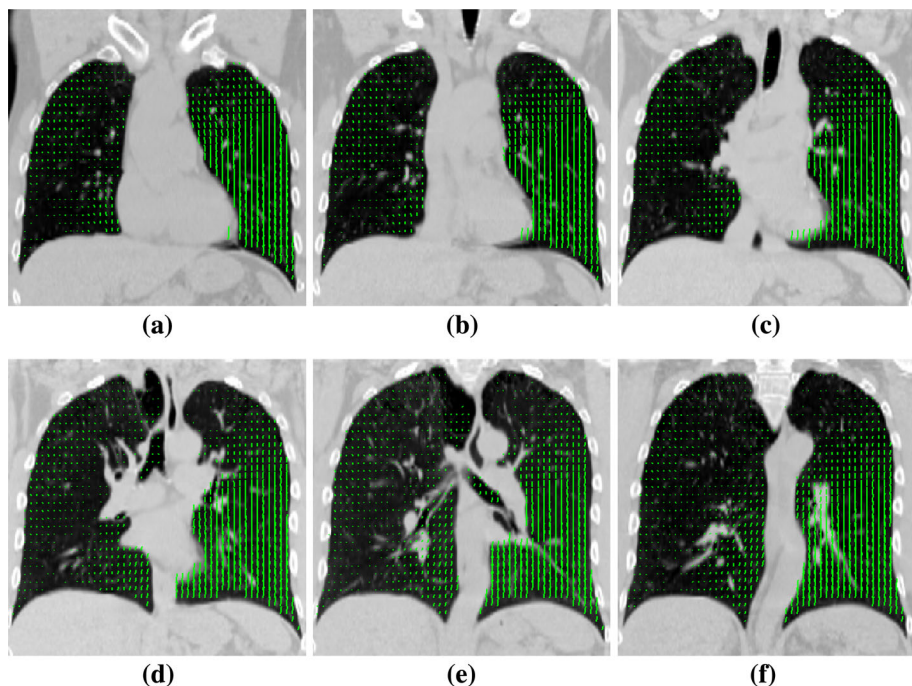


**Fig. 11** Lung CT image registration (CT 1). **a** Lung at time = 00, **b** lung at time = 50. **c** registration result

image (at time  $t = 50$ ) as the moving image and the reference image respectively. This provides the maximum displacement of the landmarks located within the lung CT images. To demonstrate the independence of our algorithm to the number of landmark points, 300 prescribed feature correspondences are enforced. Figure 11a, b show the lung CT images at time  $t = 00$  and  $t = 50$ . The image dimension of this dataset is  $256 \times 256 \times 112$ . Since the multi-grid method is applied to obtain a preconditioner to solve the f-subproblem, a linear interpolation on the image is firstly done to get the position of the landmarks corresponding to the dimension  $256 \times 256 \times 128$ , in which every dimensions has grid spacing equals to the power of 2. The 300 prescribed landmark correspondences between the two images are shown



**Fig. 12** Lung CT image registration (CT 5). **a** Lung at time = 00, **b** lung at time = 50, **c** registration result

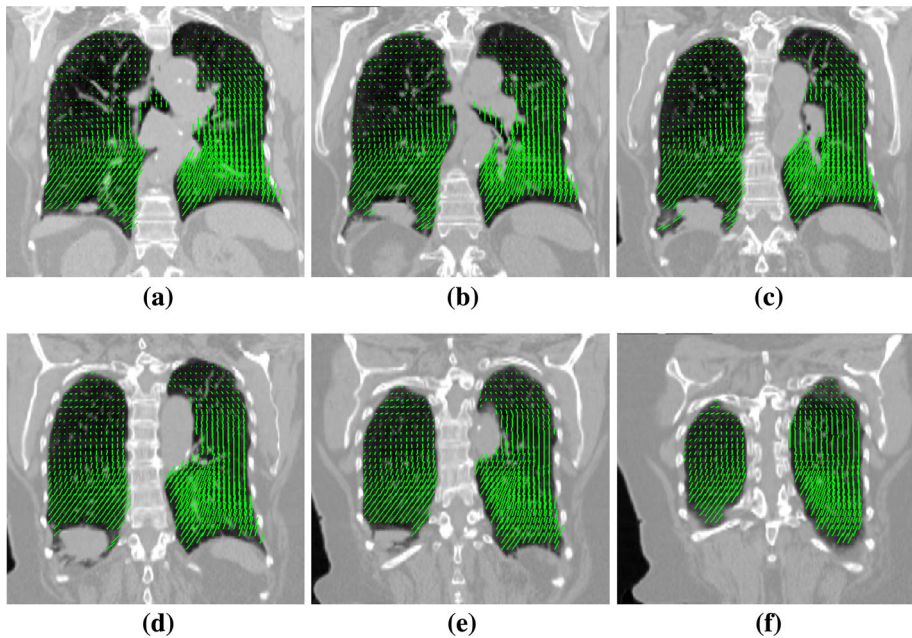


**Fig. 13** Vector field of the registration result (CT1). **a** X-slide 50, **b** X-slide 55, **c** X-slide 60, **d** X-slide 65, **e** X-slide 70, **f** X-slide 75

as the red and blue dots in the figures. Using the proposed algorithm, the landmark-based image registration of the lung CT images can be computed, which is shown in Fig. 11c.

Figure 12a, b show another set of lung CT images at time  $t = 00$  and  $t = 50$ . The image dimension of this dataset is  $512 \times 512 \times 128$ . The 300 prescribed landmark correspondences are shown as the red and blue dots in the figures. The obtained landmark-based image registration of the lung CT images is shown in Fig. 12c.

Figures 13a–f and 14a–f show the vector fields of the lung deformations obtained from the registration results. The images are the slides on the X-axis with slide numbers 50, 55, 60, 65, 70 and 75 respectively. The vector fields located inside the lung are projected



**Fig. 14** Vector field of the registration result (CT5). **a** X-slide 50, **b** X-slide 55, **c** X-slide 60, **d** X-slide 65, **e** X-slide 70, **f** X-slide 75

to the YZ planes and are visualized as green arrows in the figures. The vector fields are smooth, showing that our proposed algorithm can produce smooth landmark-based registration result.

### 5.3 Quantitative Measurements

Table 1 lists the quantitative measurements of the landmark-matching transformation obtained from the proposed algorithm and the TPS method. For a fair comparison, we first normalize the domain  $\Omega$  in each example to be the unit cube. The maximum and minimum displacement of the prescribed landmark correspondences are denoted as  $LM_{\max}$  and  $LM_{\min}$  respectively. The quantities  $e_{\max}$  and  $e_{\min}$  shows the maximum and minimum landmark mismatching error in the L-2 sense.

The maximum of the resulting conformality distortion is denoted by  $\max K$ . Note that  $\max K$  obtained from our proposed algorithm are all finite. This implies that the computed transformations in all examples are orientation-preserving. However, results generated by the TPS in the five synthetic examples have infinite value of  $\max K$ , which indicates folding occurs in the mapping obtained from TPS. For lung registration examples (CT1 - CT5), we observe that the  $\max K$  of TPS is relatively smaller than that of the proposed algorithm. This is mainly due to the inexact alignment of the landmark points by TPS which provides more freedom for the optimization of the transformation.  $\min \text{Det}$ , which is the minimum of the Jacobian, is another indicator showing the diffeomorphic property of the mapping [8,9]. #Fold counts the number of tetrahedra in which the obtained transformation has negative Jacobian. We observe that foldings occur in TPS method when the landmark displacement is large. For our proposed algorithm, no foldings are observed for both synthetic and the lung registration examples. This shows the capability of the generalized conformality distortion  $K(f)$  in enforcing the bijectivity of the transformation.

The computation time for both algorithms is also reported in the table. With the proposed numerical method applied in the algorithm, the time required for large deformation is quite reasonable (less than 30 s for sparse grids and less than four minutes for dense grid size).

## 6 Conclusion

This paper presents a new method to obtain folding-free landmark-matching transformation between general  $n$ -dimensional Euclidean spaces with large deformations. The basic idea is to extend the 2-dimensional quasi-conformal theories to general  $n$ -dimensional spaces. Given a set of landmark constraints, our goal is to look for an optimal transformation that matches landmarks. In this paper, we introduce a notion of conformality distortion of a diffeomorphism of the  $n$ -dimensional Euclidean space. The conformality distortion measures the distortion of an infinitesimal ball to an infinitesimal ellipsoid under the diffeomorphism. Our problem can then be modelled as a minimization problem of an energy functional involving the conformality term and a smoothness term. The conformality term allows the algorithm to produce folding-free transformation with minimized local geometric distortions, even with very large deformations. Alternating direction method of multipliers (ADMM) is applied in this paper to solve the optimization problem. The algorithm only involves solving an elliptic problem and a tetrahedron-wise minimization problem. Preconditioned conjugate gradient method with multi-grid V-cycle preconditioner is applied to one of the subproblem, while a fixed-point iteration is used for another subproblem. The time complexity and robustness of the algorithm is independent of the number of landmark constraints. Experimental results show that our proposed algorithm is effective for computing folding-free landmark-matching transformation, even with large number of landmarks or large deformations. In the future, we will test the algorithm on other real medical data, such as 3D MRI scan with DTI fibre tracks as the interior landmark constraints.

**Acknowledgments** The authors acknowledge the freely available lung CT data from the Deformable Image722 Registration Laboratory ([www.dir-lab.com](http://www.dir-lab.com)). Lok Ming Lui is supported by RGC GRF (CUHK Project ID: 2130363; Reference: 402413).

## References

1. Avants, B.B., Epstein, C.L., Grossman, M., Gee, J.C.: Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Med. Image Anal.* **12**(1), 26–41 (2008)
2. Bookstein, F.L.: Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* **11**(6), 567–585 (1989)
3. Borwein, J.M., Vanderwerff, J.D.: *Convex Functions: Constructions, Characterizations and Counterexamples*, vol. 109. Cambridge University Press, Cambridge (2010)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Regist. Mach. Learn.* **3**(1), 1–122 (2011)
5. Briggs, W.L., Henson, V.E., McCormick, S.F.: *A Multigrid Tutorial*. SIAM (2000)
6. La Cascio, M., Sclaroff, S., Athitsos, V.: Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3d models. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(4), 322–336 (2000)
7. Chanwimaluang, T., Fan, G., Fransen, S.R.: Hybrid retinal image registration. *IEEE Trans. Inf. Technol. Biomed.* **10**(1), 129–142 (2006)
8. Christensen, G.E., Johnson, H.J.: Consistent image registration. *IEEE Trans. Med. Imaging* **20**(7), 568–582 (2001)



9. Christensen, G.E., Rabbitt, R.D., Miller, M.I.: Deformable templates using large deformation kinematics. *IEEE Trans. Image Process.* **5**(10), 1435–1447 (1996)
10. Dupuis, P., Grenander, U., Miller, M.I.: Variational problems on flows of diffeomorphisms for image matching. *Q. Appl. Math.* **56**(3), 587–600 (1998)
11. Erikson, A.P., Astrom, K.: On the bijectivity of thin-plate splines. *Anal. Sci. Eng. Beyond Springer Proc. Math.* **6**, 93–141 (2012)
12. Gardiner, F.P., Lakic, N.: Quasiconformal Teichmüller Theory. *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI (2000)
13. Ghadimi, E., Teixeira, A., Shames, I., Johansson, M.: Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems. *IEEE Trans. Autom. Control* **60**(3), 644–658 (2015)
14. Glaunes, J., Qiu, A., Miller, M.I., Younes, L.: Large deformation diffeomorphic metric curve mapping. *Int. J. Comput. Vis.* **80**(3), 317–336 (2008)
15. Glaunes, J., Trounev, A., Younes, L.: Diffeomorphic matching of distributions: a new approach for unlabelled point-sets and sub-manifolds matching. In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference*, vol. 2, pp. 712–718 (2004)
16. Glaunes, J., Vaillant, M., Miller, M.I.: Landmark matching via large deformation diffeomorphisms on the sphere. *J. Math. Imaging Vis.* **20**(8), 179–200 (2004)
17. Glocker, B., Sotiras, A., Komodakis, N., Paragios, N.: Deformable medical image registration: setting the state of the art with discrete methods. *Ann. Rev. Biomed. Eng.* **13**, 219–244 (2011)
18. Glowinski, R., Le Tallec, P.: *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, vol. 9. SIAM, Philadelphia, PA (1989)
19. Glowinski, R., Marrocco, A.: Sur l'approximation par elements finis d'ordre un, et la resolution par penalisation-dualite d'une classe de problemes de dirichlet nonlineaires. *Rev. Francaise d'Aut. Inf. Rech. Oper.* **R-2**, 41–76 (1975)
20. Gower, J.C., Dijksterhuis, G.B.: *Procrustes Problems*, vol. 3. Oxford University Press, Oxford (2004)
21. Guo, H., Rangarajan, A., Joshi, S.: Diffeomorphic point matching. In: Paragios, N., Chen, Y., Faugeras, O (eds.) *Handbook of Mathematical Models in Computer Vision*, pp. 205–219. Springer-Verlag, New York (2006)
22. Haker, S., Angenent, S., Tannenbaum, A., Kikinis, R., Sapiro, G.: Conformal surface parameterization for texture mapping. *IEEE Trans. Vis. Comput. Graph.* **6**(2), 181–189 (2000)
23. He, J., Christensen, G.E.: Large deformation inverse consistent elastic image registration. In: *Information Processing in Medical Imaging*, vol. 2723, pp. 438–449 (2003)
24. Heckbert, P.S.: Survey of texture mapping. *IEEE Comput. Graph. Appl.* **6**(11), 56–67 (1986)
25. Hill, D.L., Batchelor, P.G., Holden, M., Hawkes, D.J.: Medical image registration. *Phys. Med. Biol.* **46**(1), R1 (2001)
26. Huang, X., Sun, Y., Metaxas, D., Sauer, F., Xu, C.: Hybrid image registration based on configural matching of scale-invariant salient region features. In: *IEEE on Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04*, vol. 4, pp. 167–167 (2004)
27. Johnson, H.J., Christensen, G.E.: Consistent landmark and intensity-based image registration. *IEEE Trans. Med. Imaging* **21**(5), 450–461 (2002)
28. Joshi, S.C., Miller, M.I.: Landmark matching via large deformation diffeomorphisms. *IEEE Trans. Image Process.* **9**(8), 1357–1370 (2000)
29. Klein, S., Staring, M., Murphy, K., Viergever, M.A., Pluim, J.P.W.: Elastix: a toolbox for intensity-based medical image registration. *IEEE Trans. Med. Imaging* **29**(1), 196–205 (2010)
30. Kovalsky, S.Z., Aigerman, N., Basri, R., Lipman, Y.: Controlling singular values with semidefinite programming. *ACM TOG* **33**(4), 68 (2014)
31. Kybic, J., Unser, M.: Fast parametric elastic image registration. *IEEE Trans. Image Process.* **12**(11), 1427–1442 (2003)
32. Lam, K.C., Gu, X.F., Lui, L.M.: Genus-one surface registration via teichmuller extremal mapping. *MIC-CAI* **8675**, 25–32 (2014)
33. Lam, K.C., Lui, L.M.: Landmark and intensity based registration with large deformations via quasiconformal maps. *SIAM J. Imaging Sci.* **7**(4), 2364–2392 (2014)
34. Lehto, O., Virtanen, K.I.: *Quasiconformal Mappings in the Plane. Die Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen mit besonderer Berücksichtigung der Anwendungsgebiete*, 2nd edn. Springer, Berlin (1973)
35. Leow, A.D., Yanovsky, I., Chiang, M.C., Lee, A.D., Klunder, A.D., Lu, A., Becker, J.T., Davis, S.W., Toga, A.W., Thompson, P.M.: Statistical properties of jacobian maps and the realization of unbiased large-deformation nonlinear image registration. *IEEE Trans. Med. Imaging* **26**(6), 822–832 (2007)

36. Lin, T., Le Guyader, C., Dinov, I., Thompson, P., Toga, A., Vese, L.: Gene expression data to mouse atlas registration using a nonlinear elasticity smoother and landmark points constraints. *J. Sci. Comput.* **50**(3), 586–609 (2012)
37. Lipman, Y.: Bounded distortion mapping spaces for triangular meshes. *ACM TOG* **31**(4), 108 (2012)
38. Lui, L.M., Lam, K.C., Wong, T.W., Gu, X.F.: Texture map and video compression using beltrami representation. *SIAM J. Imaging Sci.* **6**(4), 1880–1902 (2013)
39. Lui, L.M., Thiruvenkadam, S., Wang, Y.L., Chan, T.F., Thompson, P.M.: Optimized conformal parameterization of cortical surfaces using shape based matching of landmark curves. In: *Medical Image Computing and Computer-Assisted Intervention—MICCAI*, vol 3, pp. 52–78 (2008)
40. Lui, L.M., Thiruvenkadam, S., Wang, Y.L., Thompson, P.M., Chan, T.F.: Optimized conformal surface registration with shape-based landmark matching. *SIAM J. Imaging Sci.* **3**(1), 52–78 (2010)
41. Lui, L.M., Wang, Y.L., Chan, T.F., Thompson, P.: Landmark constrained genus zero surface conformal mapping and its application to brain mapping research. *Appl. Numer. Math.* **57**(5–7), 847–858 (2007)
42. Lui, L.M., Wen, C.F.: Geometric registration of high-genus surfaces. *SIAM J. Imaging Sci.* **7**(1), 337–365 (2014)
43. Modat, M., McClelland, J., Ourselin, S.: Lung registration using the niftyreg package. In: *Medical Image Analysis for the Clinic—A Grand Challenge*, Workshop Proceedings of from MICCAI 2010, pp. 33–42 (2010)
44. Paquin, D., Levy, D., Xing, L.: Hybrid multiscale landmark and deformable image registration. *Math. Biosci. Eng.* **4**(4), 711–737 (2007)
45. Reddy, B.S., Chatterji, B.N.: An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. Image Process.* **5**(8), 1266–1271 (1996)
46. Tyrrell Rockafellar, R.: Monotone operators and the proximal point algorithm. *SIAM J. Control Optim.* **14**(5), 877–898 (1976)
47. Rohr, K., Stiehl, H.S., Sprengel, R., Buzug, T.M., Weese, J., Kuhn, M.H.: Landmark-based elastic registration using approximating thin-plate splines. *IEEE Trans. Med. Imaging* **20**(6), 526–534 (2001)
48. Saad, Y.: *Iterative Methods for Sparse Linear Systems*. SIAM (2003)
49. Sotiras, A., Davatzikos, C., Paragios, N.: Deformable medical image registration: a survey. *IEEE Trans. Med. Imaging* **32**(7), 1153–1190 (2013)
50. Tatebe, O.: The multigrid preconditioned conjugate gradient method. In: *NASA, Langley Research Center, The Sixth Copper Mountain Conference on Multigrid Methods*, vol 2, pp. 621–634 (1993)
51. Thirion, J.P.: Image matching as a diffusion process: an analogy with Maxwell’s demons. *Med. Image Anal.* **2**(3), 243–260 (1998)
52. Tosun, D., Rettmann, M.E., Prince, J.L.: Mapping techniques for aligning sulci across multiple brains. *Med. Image Anal.* **8**(3), 295–309 (2004)
53. Trottenberg, U., Oosterlee, C.W., Schuller, A.: *Multigrid*. Academic Press, San Diego, CA (2000)
54. Vaillant, M., Glaunes, J.: *Surface Matching Via Currents*. Information Processing in Medical Imaging. Springer, Berlin (2005)
55. Vannier, M.W., Marsh, J.L.: Three-dimensional imaging, surgical planning, and image-guided therapy. *Radiol. Clin. N. Am.* **34**(3), 545–563 (1996)
56. Vercauteren, T., Pennec, X., Perchant, A., Ayache, N.: Diffeomorphic demons: efficient non-parametric image registration. *Neuroimage* **45**(1), S61–S72 (2009)
57. Wang, Y.L., Lui, L.M., Chan, T.F., Thompson, P.M.: Optimization of brain conformal mapping with landmarks. *MICCAI* **3750**, 675–683 (2005)
58. Yeo, B.T.T., Sabuncu, M.R., Vercauteren, T., Ayache, N., Fischl, B., Golland, P.: Spherical demons: fast diffeomorphic landmark-free surface registration. *IEEE Trans. Medical Imaging* **29**(3), 650–668 (2010)
59. Zeng, W., Gu, X.F.: Registration for 3d surfaces with large deformations using quasi-conformal curvature flow. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 2457–2464 (2011)
60. Zhu, S., Ma, K.K.: A new diamond search algorithm for fast block-matching motion estimation. *IEEE Trans. Image Process.* **9**(2), 287–290 (2000)
61. Zitova, B., Flusser, J.: Image registration methods: a survey. *Image Vis. Comput.* **21**(11), 977–1000 (2003)