SIAM J. Imaging Sciences
Vol. 13, No. 3, pp. 1653–1691

3D Orientation-Preserving Variational Models for Accurate Image Registration*

Daoping Zhang† and Ke Chen‡

Abstract. The Beltrami coefficient from complex analysis has recently been found to provide a robust constraint for obtaining orientation-preserving and diffeomorphic transformations for registration of planar images. There exists no such concept of the Beltrami coefficient in three or higher dimensions, although a generalized theory of quasi-conformal maps in high dimensions exists. In this paper, we first propose a new algebraic measure in three dimensions (3D) that mimics the Beltrami concept in two dimensions (2D) and then propose a corresponding registration model based on it. We then establish the existence of solutions for the proposed model and further propose a converging generalized Gauss–Newton iterative method to solve the resulting nonlinear optimization problem. In addition, we also provide another two possible regularizers in 3D. Numerical experiments show that the new model can produce more accurate orientation-preserving transformations than competing state-of-the-art registration models.

Key words. orientation-preserving maps, variational model, 3D image registration, generalized Gauss–Newton method

AMS subject classifications. 65K10, 68U10, 68W01, 49M15, 90C30

DOI. 10.1137/20M1320006

---

1. Introduction.

Image registration, also called image matching, image wrapping, or image fusion, has become one of the most important tasks in the image processing domain. It aims to find an optimal geometric transformation to align the corresponding image data, which are taken at different times, from different imaging machinery, or from different perspectives. Nowadays, image registration has a wide range of applications, such as computer vision, biological imaging, remote sensing, and medical imaging [7, 27, 33, 35, 38, 44, 53]. For registering images which differ by a small deformation or by a relative simple parametric (e.g., linear) transformation, there exist many well-known and mature methods to be employed [39, 40]. Here, we consider image registration in a variational framework to cope with the more challenging task of modeling large nonparametric deformable problems, especially the question of preserving orientation in three dimensions (3D).

In general, image registration involves two or more images. By convention, we define two related monomodality images, $R : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ as the reference and $T : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ as the template, where R and T are compactly supported in Ω and d is the dimension of the images.

---

[*] Received by the editors February 18, 2020; accepted for publication (in revised form) July 1, 2020; published electronically September 24, 2020.

https://doi.org/10.1137/20M1320006

Funding: The work of the authors was supported by the EPSRC grant EP/N014499/1.

[†] Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong (dpzhang@math.cuhk.edu.hk).

[‡] Corresponding author. EPSRC Liverpool Centre for Mathematics in Healthcare, Centre for Mathematical Imaging Techniques, and Department of Mathematical Sciences, The University of Liverpool, Liverpool L69 7ZL, United Kingdom (k.chen@liv.ac.uk, http://www.liv.ac.uk/~cmchenke).

In this work, we are primarily concerned with the case $d = 3$. The aim of image registration is to find a transformation $\boldsymbol{y}(\boldsymbol{x}) : \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$ such that

$$(T \circ \boldsymbol{y})(\boldsymbol{x}) = T(\boldsymbol{y}(\boldsymbol{x}))$$

is similar to $R(\boldsymbol{x})$, where $\boldsymbol{x} = (x_1, x_2, x_3)$ and $\boldsymbol{y}(\boldsymbol{x}) = (y_1(\boldsymbol{x}), y_2(\boldsymbol{x}), y_3(\boldsymbol{x}))$. In order to measure the difference between $T(\boldsymbol{y}(\boldsymbol{x}))$ and $R(\boldsymbol{x})$, under the monomodality case, the most widely used fidelity term is the sum of squared differences (SSD) [39, 40] defined by

$$\mathcal{D}(T \circ \boldsymbol{y}, R) := \frac{1}{2}\int_\Omega (T(\boldsymbol{y}(\boldsymbol{x})) - R(\boldsymbol{x}))^2 \mathrm{d}\boldsymbol{x} = \frac{1}{2}\|T \circ \boldsymbol{y} - R\|^2, \tag{1}$$

where $\|\cdot\|^2$ denotes the square of the L^2-norm. For the multimodality case, there are some other typical distance measures, including normalized cross correlation, mutual information, normalized gradient fields [24, 26, 34, 39, 40], and especially the more recent model [45]. As the paper places greater emphasis on the quality of the transformation $\boldsymbol{y}$, the presentation is mainly for the monomodality case, but the results are applicable to the multimodality case after a change of fidelity terms.

Minimizing $\mathcal{D}(T \circ \boldsymbol{y}, R)$ for image registration is ill-posed in the sense of Hadamard since it is not sufficient to ensure the uniqueness and continuity of the solution [43]. In order to overcome this problem, regularization is indispensable. Combining the distance measure with the regularizer, we can obtain the variational model for image registration:

$$\min_{\boldsymbol{y}} \mathcal{J}(\boldsymbol{y}) = \mathcal{D}(T \circ \boldsymbol{y}), R) + \alpha \mathcal{S}(\boldsymbol{y}), \tag{2}$$

where $\mathcal{S}(\boldsymbol{y})$ is the regularizer which can rule out the unwanted solutions and $\alpha > 0$ is a positive parameter to balance these two terms.

There exist many different regularizers which lead to many nonlinear registration models, such as the elastic model [6], fluid model [12], diffusion model [19], total variation (TV) model [22], MTV (modified TV) model [13], linear curvature model [20, 21], mean curvature model [14], Gaussian curvature model [28], and total fractional-order variation model [51]. These models can produce different registration transformations since they are inspired by different physical properties [44], each having advantages in its class of problems, though not all of these models have been tested in registration of 3D images.

However, in all of these models, folding will appear when the deformation is large or the regularization parameter is small if we impose no constraint on the transformation. Few models have built-in capabilities to impose such constraints. A transformation with folding implies that the obtained transformation itself is not a valid or acceptable solution. According to the inverse function theorem, the transformation $\boldsymbol{y}$ is locally bijective when $\det \nabla \boldsymbol{y} > 0$, where $\det \nabla \boldsymbol{y}$ is the Jacobian determinant of the transformation $\boldsymbol{y}$. Hence, constraining the Jacobian determinant of $\boldsymbol{y}$ larger than 0 is a key factor to reduce or avoid folding [8, 15, 23, 25]. We know that the geometric meaning of the Jacobian determinant of the transformation is the ratio of the change of the volume. But for some applications, it is tough for users to decide the upper bound and lower bound of the Jacobian determinant of the transformation. Only controlling the Jacobian determinant of the transformation to approximate 1 sometimes will affect the accuracy of the registration [50]. Another effective way to avoid the folding is to control the Beltrami coefficient [31, 50]. The quasi-conformal theory shows that if the infinity norm of the Beltrami coefficient μ is smaller than 1, the corresponding mapping is a homeomorphism, i.e., $|\mu| < 1 \;\Leftrightarrow\; \det \nabla \boldsymbol{y} > 0$. Normally, the Beltrami coefficient is defined in the complex space, and for two dimensional (2D) image registration, we can consider the transformation as a complex mapping and control its Beltrami coefficient to get an orientation-preserving transformation. However, since the Beltrami coefficient has no definition in 3D, we cannot directly apply the notion of the Beltrami coefficient to 3D image registration. Although it is possible to construct models [32] that are based on quasi-conformal maps [18, Ch. 6], [37], there is no such definition of $|\mu|$ that satisfies $|\mu| < 1 \;\Leftrightarrow\; \det \nabla \boldsymbol{y} > 0$ and can be used in a minimization model.

In this paper, first, we propose a new measure in 3D that mimics the norm of the Beltrami coefficient in 2D and study its properties. Second, combining with regularization, we propose the new registration model which can cope with large deformation registration problems and generate orientation-preserving transformations. The existence of the solution of the proposed model is established. In addition, we also provide another two possible regularizers in 3D. Finally, an effective numerical scheme is presented, and numerical experimental results also show that the new registration model can deliver good performances and accurate transformations and can be competitive with the other state-of-the-art registration models.

The rest of the paper is organized as follows. In section 2, we briefly review related works. In section 3, we propose our new regularizer and new registration model for 3D image registration. The existence of the solution and numerical implementation are also illustrated. In section 4, another two possible regularizers are given. Numerical experimental results are shown in section 5, and finally, a conclusion is summarized in section 6.

2. Related works.

There exist several 3D variational models, though not as many as in 2D, which can produce orientation-preserving transformations for image registration. In this section, we briefly review three representative models to highlight the outstanding challenges.

2.1. Hyperelastic models.

The hyperelastic regularizer in image registration was first used by Droske and Rumpf [15] in 2004. Their formulation of type (2) takes the form

$$\mathcal{S}(\boldsymbol{y}) = \mathcal{S}^{\text{hyper}}(\boldsymbol{y}) := \int_\Omega W(\nabla \boldsymbol{y}, \operatorname{cof}\nabla \boldsymbol{y}, \det \nabla \boldsymbol{y})\mathrm{d}\boldsymbol{x}, \tag{3}$$

where $W : \mathbb{R}^{3,3} \times \mathbb{R}^{3,3} \times \mathbb{R} \to \mathbb{R}$ is a convex function. Here, $\mathcal{S}^{\text{hyper}}(\boldsymbol{y})$ is assumed to penalize volume shrinkage, i.e., $W(L, S, V) \overset{V\to 0}{\longrightarrow} \infty$. This latter assumption (or choice) will enable us to successfully control singularity sets. In [8], $W(\nabla \boldsymbol{y}, \operatorname{cof}\nabla \boldsymbol{y}, \det \nabla \boldsymbol{y})$ is defined as follows:

$$W(\nabla \boldsymbol{y}, \operatorname{cof}\nabla \boldsymbol{y}, \det \nabla \boldsymbol{y}) := \alpha_l \phi_l(\nabla \boldsymbol{y}) + \alpha_s \phi_s(\operatorname{cof}\nabla \boldsymbol{y}) + \alpha_v \phi_v(\det \nabla \boldsymbol{y}), \tag{4}$$

where $\phi_l(X) = \|X - I\|_{\mathrm{F}}^2/2$, $\phi_s(X) = \max\{\|X\|_{\mathrm{F}}^2 - 3, 0\}^2/2$, $\phi_v(x) = ((x-1)^2/x)^2$, and $\|\cdot\|_F$ denotes the Frobenius norm. Here, since $\phi_v(x)$ goes to ∞ when v goes to 0 or ∞ and $\phi_v(x) = \phi_v(1/x)$, $\phi_v(x)$ controls the volume such that shrinkage and growth have the same price. Hence, $\mathcal{S}^{\text{hyper}}$ restricts the Jacobian determinant of the transformation $\boldsymbol{y}$ close to 1, which is suitable for certain applications (such as functional MRIs) but is too strong as a constraint in other applications (e.g., [50] shows an example in 2D).

2.2. LDDMM. The variational formulation of large deformation diffeomorphic metric mapping (LDDMM) [4, 16, 36, 48] is a widely used technique for image registration, defined by

$$\begin{aligned} &\min_{\mathcal{T},v} \mathcal{D}(\mathcal{T}(\cdot,1),R)+\alpha\mathcal{S}(v) \\ &\text{s.t. } \ \partial_t\mathcal{T}(\boldsymbol{x},t)+v(\boldsymbol{x},t)\cdot\nabla\mathcal{T}(\boldsymbol{x},t)=0 \ \text{ and } \ \mathcal{T}(\boldsymbol{x},0)=T, \end{aligned} \tag{5}$$

where $v:\Omega\times[0,1]\to\mathbb{R}^3$ is the velocity and $\mathcal{T}:\Omega\times[0,1]\to\mathbb{R}$ is a series of images. Here, LDDMM regularizes the velocity v, and we can compute its corresponding transformation $\boldsymbol{y}$ by using the information of v. When v is sufficiently smooth, it can lead to a diffeomorphic transformation $\boldsymbol{y}$, namely $\det\nabla\boldsymbol{y}>0$. However, since LDDMM involves the transport equation, the time t is introduced, and the dimension of the original problem is increased. Hence, designing an efficient solver for LDDMM is highly nontrivial; this fact is also observed in a more recent study [9].

2.3. LLL model. Lee, Lam, and Lui (LLL) [32] proposed the notion of a standard conformality distortion (see also [18, 37]) for a mapping in $\mathbb{R}^2$ to $\mathbb{R}^n(n\geq 3)$ and used it to define a variational model involving this distortion to deal with the landmark-matching problem in higher dimensional spaces. Before presenting this notion of the conformality distortion in $\mathbb{R}^n$, we first review the fundamental theory of quasi-conformal mapping.

Definition 1. *A complex map $f(z):\ \mathbb{C}\to\mathbb{C}$ is quasi-conformal if it has continuous partial derivatives and satisfies the following Beltrami equation:*

$$\frac{\partial f}{\partial\bar{z}}=\mu(z)\frac{\partial f}{\partial z} \tag{6}$$

for some complex-valued Lebesgue measurable $\mu(z):\mathbb{C}\to\mathbb{C}$ satisfying $\|\mu\|_\infty<1$, where μ is called the Beltrami coefficient [5], $2\frac{\partial f}{\partial z}=\frac{\partial f}{\partial x_1}-\boldsymbol{i}\frac{\partial f}{\partial x_2}$, *and* $2\frac{\partial f}{\partial\bar{z}}=\frac{\partial f}{\partial x_1}+\boldsymbol{i}\frac{\partial f}{\partial x_2}$ *at* $z=x_1+\boldsymbol{i}x_2$.

Here $f(z)=y_1(x_1,x_2)+\boldsymbol{i}y_2(x_1,x_2)$ links a complex map to our transformation $\boldsymbol{y}$.

Consider a simple linear map of the complex form $f(z)=az+b\bar{z}$, with complex constants a and b. If f is orientation-preserving, then the determinant is $|a|^2-|b|^2>0$ and the formulae can be rewritten as $f(z)=a(z+\mu\bar{z})$, where the complex number $\mu=b/a$ is the Beltrami coefficient; $|a|^2-|b|^2>0$ means that $|\mu|<1$. In this form, f is the stretch map $S(z)=z+\mu\bar{z}$ post-composed by a multiplication of a (which is conformal and consists of a rotation through the angle $\arg a$ and magnification by the factor $|a|$). The distortion caused by f is expressed by μ, and, from it, we can find that the angle of maximal magnification is $(\arg\mu)/2$ with magnifying factor $1+|\mu|$, and the angle of maximal shrinking is the orthogonal angle $(\arg\mu-\pi)/2$ with shrinking factor $1-|\mu|$. Naturally, motivated by this simple example, we can define by K_d the dilatation

$$K_d(f)=\frac{1+|\mu|}{1-|\mu|} \tag{7}$$

to express the ratio of the largest singular value of the Jacobian of f divided by the smallest singular value.

The dilatation K_d in (7) cannot be directly used in nD ($n \geq 3$) since the Beltrami coefficient is not defined in nD ($n \geq 3$). To find a quantity in nD resembling K_d of (7) in 2D, we start with the nD conformal mapping. For nD, let $\boldsymbol{f}(x_1, \ldots, x_n) = (y_1(x_1, \ldots, x_n), \ldots, y_n(x_1, \ldots, x_n))$ with $\nabla \boldsymbol{f}$ its $n \times n$ Jacobian matrix. An orientation-preserving condition [29] for mapping $\boldsymbol{f}$ to be conformal is

$$\nabla \boldsymbol{f}^T \nabla \boldsymbol{f} = (\det \nabla \boldsymbol{f})^{2/n} I, \tag{8}$$

where I is the identity matrix. Suppose that λ_j's with $0 \leq \lambda_1 \leq \cdots \leq \lambda_n$ are the eigenvalues of $\nabla \boldsymbol{f}^T \nabla \boldsymbol{f}$. Then we have $\|\nabla \boldsymbol{f}\|_F^2 = \lambda_1 + \cdots + \lambda_n$ and $\det \nabla \boldsymbol{f} = (\lambda_1 \cdots \lambda_n)^{1/2}$. By the eigendecomposition of $\nabla \boldsymbol{f}^T \nabla \boldsymbol{f}$, we know that (8) holds if and only if $\lambda_1 = \cdots = \lambda_n$.

In addition, by the inequality of arithmetic and geometric means and noting $\lambda_j \geq 0$ $(j = 1 \ldots, n)$, we have

$$(\lambda_1 \cdots \lambda_n)^{1/n} \leq \frac{\lambda_1 + \cdots + \lambda_n}{n} \quad \text{or} \quad \frac{1}{n}\left(\frac{\lambda_1 + \cdots + \lambda_n}{(\lambda_1 \cdots \lambda_n)^{1/n}}\right) \geq 1, \quad \text{i.e.,} \quad \frac{1}{n}\left(\frac{\|\nabla \boldsymbol{f}\|_F^2}{(\det \nabla \boldsymbol{f})^{2/n}}\right) \geq 1, \tag{9}$$

where the sign of equalities holds if and only if $\lambda_1 = \cdots = \lambda_n$. Combining these discussions, we can see that

$$\boldsymbol{f} \text{ is a conformal mapping} \quad \Longleftrightarrow \quad \tfrac{1}{n}\left(\tfrac{\|\nabla \boldsymbol{f}\|_F^2}{(\det \nabla \boldsymbol{f})^{2/n}}\right) = 1.$$

Here comes the key idea. Since we aim for a quasi-conformal mapping, minimizing the quantity $\frac{\|\nabla \boldsymbol{f}\|_F^2}{(\det \nabla \boldsymbol{f})^{2/n}}$ makes sense as it measures how far away a mapping $\boldsymbol{f}$ is from conformality. This idea is used in [32], where this quantity motivates the definition of a (generalized) conformality distortion $K(\boldsymbol{f})$ in nD:

$$K(\boldsymbol{f}) := \begin{cases} \frac{1}{n}\left(\frac{\|\nabla \boldsymbol{f}\|_F^2}{(\det \nabla \boldsymbol{f})^{2/n}}\right) & \text{if } \det \nabla \boldsymbol{f} > 0, \\ +\infty & \text{otherwise.} \end{cases} \tag{10}$$

To connect $K(\boldsymbol{f})$ to K_d in (7) for $n = 2$, note that the norm of the Beltrami coefficient μ for $\boldsymbol{f}$ is defined by

$$|\mu(\boldsymbol{f})|^2 = \frac{\|\nabla \boldsymbol{f}\|_F^2 - 2\det \nabla \boldsymbol{f}}{\|\nabla \boldsymbol{f}\|_F^2 + 2\det \nabla \boldsymbol{f}} \quad \text{or} \quad |\mu(\boldsymbol{f})|^2 = \frac{\lambda_1 + \lambda_2 - 2(\lambda_1\lambda_2)^{1/2}}{\lambda_1 + \lambda_2 + 2(\lambda_1\lambda_2)^{1/2}}, \quad \text{i.e.,} \quad |\mu(\boldsymbol{f})| = \frac{\lambda_2^{1/2} - \lambda_1^{1/2}}{\lambda_1^{1/2} + \lambda_2^{1/2}}, \tag{11}$$

where we assume all $\lambda_j > 0$. Hence (7) becomes $K_d(f) = \frac{1+|\mu|}{1-|\mu|} = (\lambda_2/\lambda_1)^{1/2}$, while (10) reduces to

$$K(\boldsymbol{f}) = \frac{1}{2}\left(\frac{\lambda_1 + \lambda_2}{(\lambda_1\lambda_2)^{1/2}}\right) = \frac{1}{2}\left((\lambda_1/\lambda_2)^{1/2} + (\lambda_2/\lambda_1)^{1/2}\right).$$

Therefore, the new $K(\boldsymbol{f})$ in (10) is equivalent to the dilatation K_d in (7) for $n = 2$, though not identical to the precise equivalence relationship from

$$K(\boldsymbol{f}) \leq K_d(f) \leq 2K(\boldsymbol{f}). \tag{12}$$

Based on this generalized conformality distortion, the LLL model is defined by

$$\min_{\boldsymbol{y}} \|K(\boldsymbol{y})\|_1 + \frac{\alpha}{2}\|\Delta \boldsymbol{y}\|_2^2 \qquad \text{s.t.} \;\; \boldsymbol{y}(p_i) = q_i, \;\; 1 \le i \le m, \tag{13}$$

where p_i and q_i are the prescribed $m \ge (n+1)$ landmark points. In (13), to get a quasi-conformal map, the first term controls the minimal conformality distortion, and the second term keeps the smoothness of the mapping with the constraints serving as the data fidelity. To implement alternative minimization iterations in the numerical solution [32], an auxiliary (matrix) variable $\boldsymbol{v} = \nabla \boldsymbol{y}$ is introduced. In the 3D case, the LLL model (13) takes the following equivalent form:

$$\min_{\boldsymbol{y}} \frac{1}{3}\left(\frac{\|\nabla \boldsymbol{y}\|_F^2}{(\det \boldsymbol{v})^{2/3}}\right) + \frac{\alpha}{2}\|\Delta \boldsymbol{y}\|_2^2 \qquad \text{s.t.} \;\; \boldsymbol{v} = \nabla \boldsymbol{y}, \;\; \det \boldsymbol{v} > 0, \;\; \text{and} \;\; \boldsymbol{y}(p_i) = q_i, \;\; 1 \le i \le m. \tag{14}$$

Then an alternating direction method with Lagrangian multipliers is applied to solve (14). Note that this model is designed for landmark registration and has no intensity information. Shortly, we will consider adapting (14) to an intensity registration framework as an option for our main model.

3. A new registration model for 3D image registration.

The starting point of our idea is to address the question of how to design a Beltrami coefficient–like quantity $|\mu|$, linking to our transformation $\boldsymbol{y}$, such that a relationship $|\mu| < 1 \Leftrightarrow \det \nabla \boldsymbol{y} > 0$ holds. Then building a new model minimizing how $|\mu|$ was used in planar cases would be immediately feasible; such a model will produce a diffeomorphic map $\boldsymbol{y}$.

However, as stated before, no such quantity $|\mu|$ exists in 3D or for $n > 3$. To fill in this gap, we first propose such a quantity in 3D and then show that it shares the same theoretical properties that a Beltrami coefficient in 2D has. Hence we shall call it a Beltrami coefficient–like distortion measure. We then employ it as a regularizer to build the new 3D model before addressing other theoretical and numerical issues.

3.1. Beltrami coefficient–like distortion measure.

Given a map $\boldsymbol{f}$ in 3D, we propose an algebraic construction to measure its departure from a conformal map.

Definition 2. *If the map* $\boldsymbol{f}(x_1, x_2, x_3) = (y_1(x_1,x_2,x_3), y_2(x_1,x_2,x_3), y_3(x_1,x_2,x_3))$ *is continuously differentiable, then we define*

$$\mathcal{N}(\boldsymbol{f}) = \frac{\|\nabla \boldsymbol{f}\|_F - \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3}}{\|\nabla \boldsymbol{f}\|_F + \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3}} \tag{15}$$

as a new algebraic measure for $\boldsymbol{f}$ *whenever* $\|\nabla \boldsymbol{f}\|_F \neq 0$.

We note that in (15), $\det \nabla \boldsymbol{f}$ can take any sign while the condition $\|\nabla \boldsymbol{f}\|_F \neq 0$ is usually satisfied in image registration because $\det \nabla \boldsymbol{f} = 1$ where there are no deformations. Then, to see how $\mathcal{N}(\boldsymbol{f})$ could be related to distortion of a conformal map, we have the following results.

Lemma 3. *The quantity* $\mathcal{N}$ *defined by* (15) *for a map* $\boldsymbol{f}$ *possesses the following properties:*
P1 *If* $\mathcal{N}(\boldsymbol{f}) = 0$*, then all the singular values of* $\nabla \boldsymbol{f}$ *are equal.*

P2 *$\mathcal{N}$ is nonnegative:* $0 \leq \mathcal{N}(\boldsymbol{f}) \leq \infty$.
P3 *The Beltrami coefficient–like equivalence holds:* $\mathcal{N}(\boldsymbol{f}) < 1 \Leftrightarrow \det \nabla \boldsymbol{f} > 0$.
P4 *The special case holds:* $\mathcal{N}(\boldsymbol{f}) = 1 \Leftrightarrow \det \nabla \boldsymbol{f} = 0$.
P5 $1 < \mathcal{N}(\boldsymbol{f}) \leq \infty \Leftrightarrow \det \nabla \boldsymbol{f} < 0$.
P6 *If the singular values of $\nabla \boldsymbol{f}$ are equal, then $\mathcal{N}(\boldsymbol{f}) = 0$ when $\det \nabla \boldsymbol{f} > 0$, and $\mathcal{N}(\boldsymbol{f})$ is ∞ when $\det \nabla \boldsymbol{f} < 0$.*
P7 *The quantity $\mathcal{N}$ is invariant under the scalar multiplication and rigid-body motion actions.*

Proof. For P1, if $\mathcal{N}(\boldsymbol{f}) = 0$, according to (15), $\|\nabla \boldsymbol{f}\|_F = \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3}$. Hence, $\det \nabla \boldsymbol{f}$ is nonnegative. Let σ_1, σ_2, and σ_3 be the singular values of $\nabla \boldsymbol{f}$. Then we have $\|\nabla \boldsymbol{f}\|_F = \sqrt{\sum_{i=1}^3 \sigma_i^2}$ and $\det \nabla \boldsymbol{f} = \Pi_{i=1}^3 \sigma_i$. By (9), we have $\sqrt{\sum_{i=1}^3 \lambda_i} \geq \sqrt{3}\sqrt{(\Pi_{i=1}^3 \lambda_i)^{1/3}}$. Since $\lambda_i = \sigma_i^2$ for $i = 1, 2, 3$, we further have $\sqrt{\sum_{i=1}^3 \sigma_i^2} \geq \sqrt{3}(\Pi_{i=1}^3 \sigma_i)^{1/3}$. The equality holds if and only if $\sigma_1 = \sigma_2 = \sigma_3$.

For P2, if $\det \nabla \boldsymbol{f}$ is positive, then the denominator is obviously nonnegative. By the application of (9) as in P1, we have $\|\nabla \boldsymbol{f}\|_F \geq \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3}$, so the numerator is nonnegative. Similarly, if $\det \nabla \boldsymbol{f}$ is negative, the numerator and denominator are both nonnegative. Hence, we have $0 \leq \mathcal{N}(\boldsymbol{f}) \leq \infty$.

P3 – P5 directly follows from (15).

For P6, if $\sigma_1 = \sigma_2 = \sigma_3$, when $\det \nabla \boldsymbol{f} > 0$, we have $\|\nabla \boldsymbol{f}\|_F - \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3} = 0$ and $\|\nabla \boldsymbol{f}\|_F + \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3} = 2\|\nabla \boldsymbol{f}\|_F$; then $\mathcal{N}(\boldsymbol{f}) = 0$. But when $\det \nabla \boldsymbol{f} < 0$, since $\det \nabla \boldsymbol{f} = -\Pi_{i=1}^3 \sigma_i$, we have $\|\nabla \boldsymbol{f}\|_F + \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3} = 0$ and $\|\nabla \boldsymbol{f}\|_F - \sqrt{3}(\det \nabla \boldsymbol{f})^{1/3} = 2\|\nabla \boldsymbol{f}\|_F$; then $\mathcal{N}(\boldsymbol{f}) = \infty$.

For P7, $c\sigma_1, c\sigma_2, c\sigma_3$ are the singular values of $\nabla c\boldsymbol{f}$ for any $c > 0$ since $\sigma_1, \sigma_2, \sigma_3$ are singular values of $\nabla \boldsymbol{f}$. So we have $\|\nabla c\boldsymbol{f}\|_F = c\sqrt{\sum_{i=1}^3 \sigma_i^2}$ and $\det \nabla c\boldsymbol{f} = c^3 \Pi_{i=1}^3 \sigma_i$ and $\mathcal{N}(\boldsymbol{f}) = \mathcal{N}(c\boldsymbol{f})$. In addition, if O is an orthogonal matrix and b is a translation, then the Jacobian of $\boldsymbol{f}(O\boldsymbol{x} + b)$ is $O^T \nabla \boldsymbol{f}$. Since the singular values of $\nabla \boldsymbol{f}$ and $O^T \nabla \boldsymbol{f}$ are the same, we have $\mathcal{N}(\boldsymbol{f}(\boldsymbol{x})) = \mathcal{N}(\boldsymbol{f}(O\boldsymbol{x} + b))$. Hence, the quantity $\mathcal{N}$ is invariant under the scalar multiplication and rigid-body motion actions. ■

We remark that property P5 from Lemma 3, i.e., $\mathcal{N}(\boldsymbol{y}) < 1 \Leftrightarrow \det \nabla \boldsymbol{y} > 0$, is our expectation for $\mathcal{N}(\boldsymbol{y})$ in 3D to inherit the key property of the Beltrami coefficient in 2D: $|\mu| < 1 \Leftrightarrow \det \nabla \boldsymbol{y} > 0$. Hence, by analogy, we may view $\mathcal{N}(\boldsymbol{f})$ as a measure of distortion on conformality. The generalization given by [1] for quasi-conformal maps in 3D is very interesting, but their quasi-conformal dilatation is nondifferentiable, so it cannot be easily adapted to a variational model. We can build a 3D variational model using $\mathcal{N}(\boldsymbol{y})$ in an unconstrained optimization framework, similar to the 2D case [50]. In fact, we can apply this result to most variational models [11] that do not yet guarantee a diffeomorphic map.

Another observation on $\mathcal{N}(\boldsymbol{y})$ is that controlling $\mathcal{N}(\boldsymbol{y})$ not only ensures the bijectivity, but also guarantees the smoothness, which means that this new regularizer is more likely to produce a regular transformation. Promoting $\mathcal{N}(\boldsymbol{y}) < 1$ does not restrict the range of the Jacobian determinant of the transformation. For example, consider two simple and separate maps: $\boldsymbol{y}_1(\boldsymbol{x}) = 0.1\boldsymbol{x} = 0.1(x_1, x_2, x_3)$ and $\boldsymbol{y}_2(\boldsymbol{x}) = 10\boldsymbol{x} = 10(x_1, x_2, x_3)$. We have $\mathcal{N}(\boldsymbol{y}_1) = \mathcal{N}(\boldsymbol{y}_2) = 0$, but $\det \nabla \boldsymbol{y}_1 = 0.001$ and $\det \nabla \boldsymbol{y}_2 = 1000$.

Finally, for completeness, we may extend our above measure (15) from 3D to $\boldsymbol{f}$ in nD (beyond $n = 3$):

$$\mathcal{N}(\boldsymbol{f}) = \begin{cases} \frac{\|\nabla \boldsymbol{f}\|_F - \sqrt{n}(\det \nabla \boldsymbol{f})^{1/n}}{\|\nabla \boldsymbol{f}\|_F + \sqrt{n}(\det \nabla \boldsymbol{f})^{1/n}} & \text{if } n \text{ is odd,} \\ \frac{\|\nabla \boldsymbol{f}\|_F - \sqrt{n}(\det \nabla \boldsymbol{f})^{1/n}}{\|\nabla \boldsymbol{f}\|_F + \sqrt{n}(\det \nabla \boldsymbol{f})^{1/n}} & \text{if } n \text{ is even and } \det \nabla \boldsymbol{f} > 0, \\ \infty & \text{otherwise.} \end{cases} \tag{16}$$

Now, we see the connection between the new measure (16) and the standard Beltrami coefficient $|\mu|$ from (11) for $n = 2$ and $\det \nabla \boldsymbol{f} > 0$. First rewrite $|\mu|^2$ from (11) as

$$|\mu(\boldsymbol{f})|^2 = \frac{\|\nabla \boldsymbol{f}\|_F^2 - 2\det \nabla \boldsymbol{f}}{\|\nabla \boldsymbol{f}\|_F^2 + 2\det \nabla \boldsymbol{f}} = \mathcal{N}(\boldsymbol{f}) \frac{(\|\nabla \boldsymbol{f}\|_F + \sqrt{2}(\det \nabla \boldsymbol{f})^{1/2})^2}{\|\nabla \boldsymbol{f}\|_F^2 + 2\det \nabla \boldsymbol{f}}.$$

Then we see that the following relationship holds (noting the close resemblance to (12)):

$$\mathcal{N}(\boldsymbol{f}) \leq |\mu(\boldsymbol{f})|^2 \leq 2\mathcal{N}(\boldsymbol{f}), \tag{17}$$

which confirms that our new measure is equivalent to the Beltrami coefficient in 2D. Importantly, for $n \geq 2$, our new measure (16) shares the key property as $|\mu(\boldsymbol{f})|^2$: $\mathcal{N}(\boldsymbol{f}) < 1 \Leftrightarrow \det \nabla \boldsymbol{f} > 0$.

3.2. A new 3D image registration model.

Here, we first formulate our new model to deal with 3D image registration problems. Equipped with the knowledge of $|\mu| < 1 \Leftrightarrow \det \nabla \boldsymbol{y} > 0$, we propose the following new variational model for 3D image registration:

$$\min_{\boldsymbol{y}} \mathcal{J}(\boldsymbol{y}) := \frac{1}{2}\int_\Omega (T \circ \boldsymbol{y} - R)^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y} - \boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_2}{2}\int_\Omega \|\nabla^2(\boldsymbol{y} - \boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \beta \int_\Omega \phi(\mathcal{N}(\boldsymbol{y}))\mathrm{d}\boldsymbol{x}, \tag{18}$$

where ∇^2 denotes the Hessian operator and we define

$$\phi(v) = \frac{v^2}{(v-1)^2 + 1}. \tag{19}$$

Other choices of ϕ, e.g., $\phi(v) = v^2$ or $\phi(v) = v^2/((v-1)^2 + 10^{-5})$, are also permitted as long as they promote $\mathcal{N} < 1$ (or $\det \nabla \boldsymbol{y} > 0$). The key message is that the resulting transformation $\boldsymbol{y}$ will be orientation-preserving under the Dirichlet boundary conditions [2].

3.3. Mathematical analysis of the proposed model (18).

Registration models are usually nonconvex with respect to $\boldsymbol{y}$, and consequently there is no uniqueness. Here we address the solution existence of the nonconvex model (18). Since the proposed model (18) involves second order derivatives, the natural solution space should be $W^{2,2}(\Omega)$.

We first consider the solution space where the determinant is essentially positive: $\mathcal{A}_1 = \{\boldsymbol{y} \in W^{2,2}(\Omega) : |\int_\Omega \boldsymbol{y}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}| \leq |\Omega|\,(C_1 + \mathrm{diam}(\Omega))\,, \det \nabla \boldsymbol{y} \in L^2(\Omega), \det \nabla \boldsymbol{y} > 0, \text{a.e.}\}$, motivated by [8, 15]. In fact, we shall now check if their analysis tools can be used to establish the solution existence of (18). To this end, using the notation in [8, 15], we first rewrite model (18) in the following framework:

$$\mathcal{J}(\boldsymbol{y}) = \int_\Omega \varphi_1(\boldsymbol{x}, \boldsymbol{y}, \nabla \boldsymbol{y}, \nabla^2 \boldsymbol{y}, \det \nabla \boldsymbol{y})\mathrm{d}\boldsymbol{x}, \tag{20}$$

where $\varphi_1(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta, \Psi) = \frac{1}{2}(T \circ \boldsymbol{y} - R)^2 + \frac{\alpha_1}{2}|\psi - I|^2 + \frac{\alpha_2}{2}|\Theta|^2 + \beta\phi(\frac{\|\psi\|_F - \sqrt{3}\Psi^{1/3}}{\|\psi\|_F + \sqrt{3}\Psi^{1/3}})$. Although this φ_1 is convex with respect to $\Theta = \nabla^2 \boldsymbol{y}$, clearly, φ_1 is nonconvex with respect to $\psi = \nabla \boldsymbol{y}$ and $\Psi = \det \nabla \boldsymbol{y}$. Consequently we cannot apply their analysis method by calculus of variations.

Then, to overcome this nonconvexity issue, we can rewrite (20) in the following form:

$$\mathcal{J}(\boldsymbol{y}) = \int_\Omega \varphi_2(\boldsymbol{x}, \boldsymbol{y}, \nabla \boldsymbol{y}, \nabla^2 \boldsymbol{y}, \mathcal{N}(\boldsymbol{y}))\mathrm{d}\boldsymbol{x}, \tag{21}$$

where $\varphi_2(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta, \Psi) = \frac{1}{2}(T \circ \boldsymbol{y} - R)^2 + \frac{\alpha_1}{2}|\psi - I|^2 + \frac{\alpha_2}{2}|\Theta|^2 + \beta\phi\left(\Psi\right)$. It is evident that this new φ_2 is convex with respect to ψ, Θ, Ψ. We now modify the solution set $\mathcal{A}_2 = \{\boldsymbol{y} \in W^{2,2}(\Omega) : |\int_\Omega \boldsymbol{y}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}| \le |\Omega|\left(C_1 + \mathrm{diam}(\Omega)\right), \mathcal{N}(\boldsymbol{y}) \in L^2(\Omega), \mathcal{N}(\boldsymbol{y}) < 1, \text{a.e.}\}$ and establish the coercivity of (21) with respect to the product space $\mathcal{X} = W^{2,2}(\Omega) \times L^2(\Omega)$. Building the weak lower semicontinuity of (21), i.e.,

$$(\boldsymbol{y}^k, \mathcal{N}(\boldsymbol{y}^k)) \rightharpoonup (\boldsymbol{y}, \mathcal{V}) \Longrightarrow \lim_{k\to\infty} \int_\Omega \varphi_2(\boldsymbol{x}, \boldsymbol{y}^k, \nabla \boldsymbol{y}^k, \nabla^2 \boldsymbol{y}^k, \mathcal{N}(\boldsymbol{y}^k))\mathrm{d}\boldsymbol{x} \ge \int_\Omega \varphi_2(\boldsymbol{x}, \boldsymbol{y}, \nabla \boldsymbol{y}, \nabla^2 \boldsymbol{y}, \mathcal{V})\mathrm{d}\boldsymbol{x} \tag{22}$$

is also fine. However verifying $\mathcal{V} = \mathcal{N}(\boldsymbol{y})$ is highly nontrivial, and it is not yet possible to do this due to the high nonlinearity of the new regularizer $\mathcal{N}$ from (15). We seek a different way to establish the existence.

We now consider an analysis method which does not require convexity for all main variables. Our starting point is the following result.

Lemma 4 (see [52]). *Let $\Omega \subset \mathbb{R}^3$ be an open set, and let $\rho : \Omega \times \mathbb{R}^3 \times \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3^2} \to [0, +\infty)$ satisfy the following assumptions:*

(i) *ρ is a Carathéodory function:*
  1. *$\rho(\boldsymbol{x}, \cdot, \cdot, \cdot)$ is continuous for almost every $\boldsymbol{x} \in \Omega$.*
  2. *$\rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta)$ is measurable in $\boldsymbol{x}$ for every $(\boldsymbol{y}, \psi, \Theta) \in \mathbb{R}^3 \times \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3^2}$.*

(ii) *$\rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta)$ is quasi-convex with respect to Θ.*

(iii) *$0 \le \rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta) \le a(\boldsymbol{x}) + C(|\boldsymbol{y}|^2 + |\psi|^2 + |\Theta|^2)$ for some $a(\boldsymbol{x}) \in L^1(\Omega)$, $C > 0$.*

Then $\mathcal{J}(\boldsymbol{y}) = \int_\Omega \rho(\boldsymbol{x}, \boldsymbol{y}, \nabla \boldsymbol{y}, \nabla^2 \boldsymbol{y})\mathrm{d}\boldsymbol{x}$ is weak lower semicontinuous (wlsc) in $W^{2,2}(\Omega)$.

When applying Lemma 4 to the proposed model (18), we see that the requirement on the convexity of the highest order variable (Θ) can be satisfied, but the boundedness of the objective functional with respect to other variables has to be established. For this purpose, we rewrite the energy $\mathcal{J}(\cdot)$ of (18) in the following form, which fits the setting of Lemma 4:

$$\mathcal{J}(\boldsymbol{y}) = \int_\Omega \rho(\boldsymbol{x}, \boldsymbol{y}, \nabla \boldsymbol{y}, \nabla^2 \boldsymbol{y})\mathrm{d}\boldsymbol{x}, \tag{23}$$

where $\rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta) = \frac{1}{2}(T \circ \boldsymbol{y} - R)^2 + \frac{\alpha_1}{2}|\psi - I|^2 + \frac{\alpha_2}{2}|\Theta|^2 + \beta\phi(\frac{\|\psi\|_F - \sqrt{3}(\det \psi)^{1/3}}{\|\psi\|_F + \sqrt{3}(\det \psi)^{1/3}})$. To proceed, define the solution space $\mathcal{W} = \{\boldsymbol{y} \in W^{2,2}(\Omega) : \boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{x} \text{ on } \partial\Omega\}$. We assume that the images T and R are continuous and compactly supported in Ω. Then, we have the following result.

Lemma 5. *Assume that the images T and R are continuous and compactly supported in Ω. Then we have the following:*

(i) *The functional* ρ *from* (23) *is bounded as follows (for some constants* $a, C > 0$*):*

$$0 \le \rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta) \le a + C(|\boldsymbol{y}|^2 + |\psi|^2 + |\Theta|^2).$$

(ii) *The energy functional* $\mathcal{J}(\cdot)$ *in* (23) *is wlsc in* $W^{2,2}(\Omega)$.

Proof. (i) T and R are bounded by c_1 because they are compactly supported in Ω. Since $\phi(v) = v^2/((v-1)^2+1) \le 2$ for any v (more precisely, ≤ 1 for $v \le 0$, ≤ 2 for $0 < v \le 2$, and in $(1,2)$ for $v > 2$), we have

$$\begin{aligned} \rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta) &= \frac{1}{2}(T \circ \boldsymbol{y} - R)^2 + \frac{\alpha_1}{2}|\psi - I|^2 + \frac{\alpha_2}{2}|\Theta|^2 + \beta\phi\left(\frac{\|\psi\|_F - \sqrt{3}(\det\psi)^{1/3}}{\|\psi\|_F + \sqrt{3}(\det\psi)^{1/3}}\right) \\ &\le 2c_1^2 + c_2 + \frac{\alpha_1}{2}|\psi|^2 + \frac{\alpha_2}{2}|\Theta|^2 + 2\beta \\ &\le \frac{\alpha}{2}(|\boldsymbol{y}|^2 + |\psi|^2 + |\Theta|^2) + 2c_1^2 + c_2 + 2\beta, \end{aligned} \tag{24}$$

where $\alpha = \max\{\alpha_1, \alpha_2\}$. Then, the function $\rho(\cdot)$ satisfies $0 \le \rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta) \le a + C(|\boldsymbol{y}|^2 + |\psi|^2 + |\Theta|^2)$; i.e., it fulfils condition (iii) of Lemma 4 with $a(\boldsymbol{x}) \equiv a = 2c_1^2 + c_2 + 2\beta$ and $C = \alpha/2$.

(ii) We now verify that the functional $\rho(\cdot)$ fulfils all the assumptions of Lemma 4:

- Since the T and R are continuous and $\boldsymbol{y} \in \mathcal{W}$, the function $\rho(\cdot)$ is a Carathéodory function.
- It is easy to check that $\rho(\boldsymbol{x}, \boldsymbol{y}, \psi, \Theta)$ is convex with respect to Θ, implying that it is also quasi-convex.

Then, together with (1), by Lemma 4, the energy $\mathcal{J}(\cdot)$ is wlsc in $W^{2,2}(\Omega)$. ■

We are now ready to prove the existence of a solution for the minimization model (18).

Theorem 6. *Assume that images* T *and* R *are continuous and compactly supported in* Ω. *Then the minimization problem* (18) *admits at least one solution in the space* $\mathcal{W}$.

Proof. Since $\mathcal{J}(\boldsymbol{y})$ has a lower bound 0, there exists a minimizing sequence $(\boldsymbol{y}_n)_{n\in\mathbb{N}} \subset \mathcal{W}$ of $\mathcal{J}(\cdot)$, i.e.,

$$\mathcal{J}(\boldsymbol{y}_n) \underset{n\to\infty}{\longrightarrow} m_1 := \inf_{\boldsymbol{y}\in\mathcal{W}} \mathcal{J}(\boldsymbol{y}).$$

In addition, since $\mathcal{J}(Id) = \frac{1}{2}\|T \circ \boldsymbol{y} - R\|^2$ is finite, we can assume that $(\mathcal{J}(\boldsymbol{y}_n))_{n\in\mathbb{N}}$ is bounded above by a constant $m_2 > 0$. Using the generalized Poincaré inequality and the boundary condition, there exist constants $C_1, C_2 \in \mathbb{R}$ such that

$$\mathcal{J}(\boldsymbol{y}) \ge C_1\|\boldsymbol{y}\|^2_{W^{2,2}} + C_2. \tag{25}$$

The inequality (25) guarantees that the sequence $(\boldsymbol{y}_n)_{n\in\mathbb{N}}$ is uniformly bounded in $\mathcal{W}$, i.e.,

$$m_2 \ge \mathcal{J}(\boldsymbol{y}_n) \ge C_1\|\boldsymbol{y}_n\|^2_{W^{2,2}} + C_2.$$

Since $W^{2,2}$ is a reflexive space, there exists a subsequence, denoted $(\boldsymbol{y}_{n_l})_{l\in\mathbb{N}}$, such that $\boldsymbol{y}_{n_l} \underset{l\to\infty}{\rightharpoonup} \boldsymbol{y}^*$ weakly in $W^{2,2}$. By wlsc from Lemma 5, we obtain

$$\inf_{\boldsymbol{y}\in\mathcal{W}} \mathcal{J}(\boldsymbol{y}) = \lim_{n\to\infty} \mathcal{J}(\boldsymbol{y}_n) = \lim_{l\to\infty} \mathcal{J}(\boldsymbol{y}_{n_l}) \ge \mathcal{J}(\boldsymbol{y}^*) \ge \inf_{\boldsymbol{y}\in\mathcal{W}} \mathcal{J}(\boldsymbol{y}).$$

Hence, $\boldsymbol{y}^*$ is in the space $\mathcal{W}$. ■

To explain that the positivity of the Jacobian determinant of the transformation does not intervene in the existence of the solution, we modify the admissible space $\mathcal{W}$ to a new space $\overline{\mathcal{W}} = \{\boldsymbol{y} \in W^{2,2}(\Omega) : \boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{x} \text{ on } \partial\Omega, \det \nabla \boldsymbol{y} \geq \epsilon, \text{a.e., for a small } \epsilon > 0\}$. Then we have the following theorem.

Theorem 7. *Assume that images T and R are continuous and compactly supported in Ω. Then the minimization problem (18) admits at least one solution in the space $\overline{\mathcal{W}}$.*

Proof. Here, we just need to prove that the space $\overline{\mathcal{W}}$ is weakly closed with respect to $W^{2,2}$-topology. According to the Kondrachov embedding theorem, for any p such that $3 < p < 6$, $W^{2,2}$ is compactly embedded in $W^{1,p}$. Hence, $\boldsymbol{y}_n \underset{n\to\infty}{\rightharpoonup} \boldsymbol{y}$ weakly in $W^{2,2}$ implies that $\boldsymbol{y}_n \underset{n\to\infty}{\rightarrow} \boldsymbol{y}$ strongly in $W^{1,p}$, which also shows that $\boldsymbol{y}_n \underset{n\to\infty}{\rightharpoonup} \boldsymbol{y}$ weakly in $W^{1,p}$. By the weak continuity of determinants [17, sect 8.2.4, Lemma], we have $\det \nabla \boldsymbol{y}_n \underset{n\to\infty}{\rightharpoonup} \det \nabla \boldsymbol{y}$ weakly in $L^q, q = p/3 > 1$. Then the mapping $F(\boldsymbol{y}) = \det \nabla \boldsymbol{y}$ from $\overline{\mathcal{W}}$ to L^q is continuous with respect to the weak topology on both $W^{2,2}$ and L^q. Hence, $\overline{\mathcal{W}}$ is the preimage of the closed set $\{\det \nabla \boldsymbol{y} \in L^q : \boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{x} \text{ on } \partial\Omega, \det \nabla \boldsymbol{y} \geq \epsilon, \text{a.e., for any small } \epsilon > 0\}$ under the weakly continuous mapping F with respect to the weak topology on $W^{2,2}$. Thus, $\overline{\mathcal{W}}$ is weakly closed with respect to $W^{2,2}$-topology.

Then, similar to the proof of Theorem 6, there exists $\boldsymbol{y}_{n_l} \underset{l\to\infty}{\rightharpoonup} \boldsymbol{y}^*$ weakly in $\overline{\mathcal{W}}$ such that

$$\inf_{\boldsymbol{y}\in\overline{\mathcal{W}}} \mathcal{J}(\boldsymbol{y}) = \lim_{n\to\infty} \mathcal{J}(\boldsymbol{y}_n) = \lim_{l\to\infty} \mathcal{J}(\boldsymbol{y}_{n_l}) \geq \mathcal{J}(\boldsymbol{y}^*) \geq \inf_{\boldsymbol{y}\in\overline{\mathcal{W}}} \mathcal{J}(\boldsymbol{y})$$

and $\boldsymbol{y}^*$ is a minimizer in the space $\overline{\mathcal{W}}$. ■

Remark 1. In practice, we do not need this space $\overline{\mathcal{W}}$ nor do we need to add the constraint $\det \nabla \boldsymbol{y} > 0$ since a suitably large β in our model will ensure the one-to-one transformation.

3.4. A convergent numerical algorithm. There are two possible approaches to solving a variational model such as the proposed (18). One is the partial differential equation approach: first derive the Euler–Lagrange equation and then solve it numerically. Here, we consider the other approach of optimization, the first-discretize-then-optimize method, to solve our model (18).

First, we choose a suitable discrete scheme to discretize the variational model (18) to derive a finite-dimensional optimization problem. Then, we choose an optimization method to solve the resulting unconstrained optimization problem. Two popular methods are the alternating direction method of multipliers method (ADMM) [49] and the Gauss–Newton method [11]. Here we take the latter method (and briefly discuss the former later).

Discretization. We discretize our proposed model (18) on the spatial domain $\Omega = [0, \omega_1] \times [0, \omega_2] \times [0, \omega_3]$. In the implementation, we employ the nodal grid (Figure 1) and define a spatial partition

(26)
$$\Omega_h^n = \{\boldsymbol{x}^{i,j,k} \in \Omega \mid \boldsymbol{x}^{i,j,k} = (x_1^i, x_2^j, x_3^k) = (ih_1, jh_2, kh_3), 0 \leq i \leq n_1, 0 \leq j \leq n_2, 0 \leq k \leq n_3\},$$

where $h_l = \frac{\omega_l}{n_l}, 1 \leq l \leq 3$, and the discrete domain consists of $n_1 n_2 n_3$ cells of size $h_1 \times h_2 \times h_3$. We discretize the transformation field $\boldsymbol{y}$ on the nodal grid, namely $\boldsymbol{y}^{i,j,k} = (y_1^{i,j,k}, y_2^{i,j,k}, y_3^{i,j,k}) =$

$(y_1(\boldsymbol{x}^{i,j,k}), y_2(\boldsymbol{x}^{i,j,k}), y_3(\boldsymbol{x}^{i,j,k}))$. In order to simplify the presentation, we denote

$$h = h_1 h_2 h_3, \quad N = n_1 n_2 n_3, \quad N_1 = (n_1+1)(n_2+1)(n_3+1), \tag{27}$$

and, according to the lexicographical ordering, we reshape

$$X = (x_1^0, \ldots, x_1^{n_1}, x_2^0, \ldots, x_2^{n_2}, x_3^0, \ldots, x_3^{n_3})^T \in \mathbb{R}^{3N_1}$$

and

$$Y = (y_1^{0,0,0}, \ldots, y_1^{n_1,n_2,n_3}, y_2^{0,0,0}, \ldots, y_2^{n_1,n_2,n_3}, y_3^{0,0,0}, \ldots, y_3^{n_1,n_2,n_3})^T \in \mathbb{R}^{3N_1}.$$

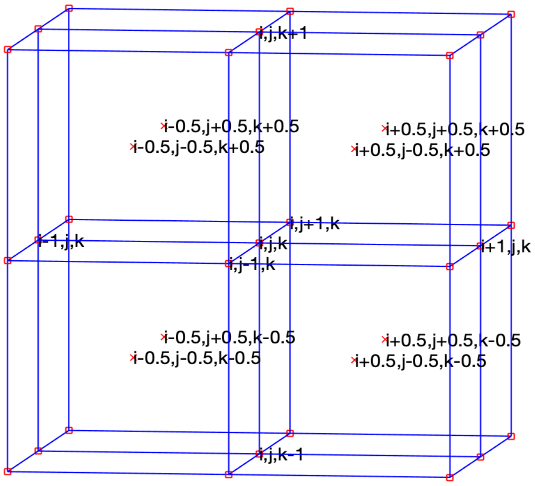


Figure 1. *Partition of the domain* Ω*. Nodal grid* $\square$ *and cell-centered grid* $\times$*.*

For the fitting term in (18), according to the cell-centered partition and midpoint rule, we get the following approximation:

$$\mathcal{D}(T \circ \boldsymbol{y}, R) := \frac{1}{2}\int_\Omega (T \circ \boldsymbol{y} - R)^2 \mathrm{d}\boldsymbol{x} \approx \frac{h}{2}(\vec{T}(PY) - \vec{R})^T(\vec{T}(PY) - \vec{R}). \tag{28}$$

Here, $\vec{R} = \vec{R}(PX) \in \mathbb{R}^N$ is the discretized reference image and $\vec{T}(PY) \in \mathbb{R}^N$ is the discretized deformed template image, where $P \in \mathbb{R}^{3N \times 3N_1}$ is an averaging matrix from the nodal grid to the cell-centered grid [23, 25].

Based on the forward difference and midpoint rule, for the first order regularizer in (18), we have the following approximation:

$$\mathcal{S}_1(\boldsymbol{y}) := \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y} - \boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} \approx \frac{\alpha_1 h}{2}(Y - X)^T A^T A (Y - X), \tag{29}$$

where A is as derived in Appendix A.

Remark 2. For (29), we have used the forward difference $\partial_{x_1} y_1^{i,j,k} \approx (y_1^{i+1,j,k} - y_1^{i,j,k})/h_1$ in Appendix A. Although the long stencil $\partial_{x_1} y_1^{i,j,k} \approx (y_1^{i+1,j,k} - y_1^{i-1,j,k})/(2h_1)$ yields second order accuracy, it is not recommended because for the high oscillatory input $[0; 1; 0; 1; \dots ; 1; 0]$, this stencil will lead to a zero derivative [40].

Based on the second order cell-centered difference and midpoint rule, for the second order regularizer in (18), we have the following approximation:

$$\mathcal{S}_2(\boldsymbol{y}) := \frac{\alpha_2}{2} \int_\Omega \|\nabla^2(\boldsymbol{y} - \boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} \approx \frac{\alpha_2 h}{2} (Y - X)^T B^T B (Y - X), \tag{30}$$

where B is as derived in Appendix B.

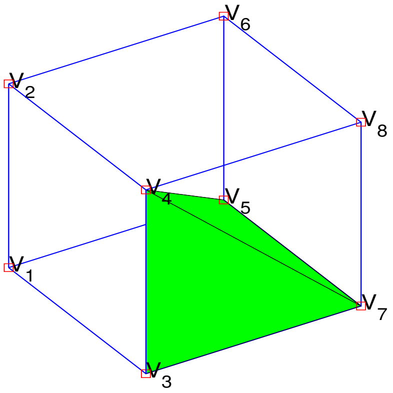


Figure 2. *Partition of a voxel.* $V_1, \dots, V_8$ *are vertices.*

Since our new regularizer $\mathcal{N}(\boldsymbol{y})$ involves $\det \nabla \boldsymbol{y}$, we should choose a suitable discretization to ensure $\det \nabla \boldsymbol{y} > 0$ when there is no mesh folding. Finite difference approximations using six neighboring pixels cannot detect folding; namely, even when the mesh has folding, $\det \nabla \boldsymbol{y}$ may still be positive. A good solution is to construct local finite elements based on a large stencil and then compute $\det \nabla \boldsymbol{y}$, since (as pointed out in [26]) a tetrahedron cannot twist unless its volume changes sign. In addition, [8] ensures the regularity of various partitions of a voxel. Hence, we divide each voxel into six tetrahedrons $(V_3V_7V_4V_5, V_3V_1V_4V_5, V_4V_1V_2V_5, V_7V_4V_5V_8, V_4V_5V_8V_6, V_4V_2V_5V_6)$ (see Figure 2), and in each tetrahedron, we use three linear interpolating functions to approximate y_1, y_2, and y_3, respectively.

According to this partition, we can get

$$\mathcal{S}_{\text{New}}(\boldsymbol{y}) = \beta \int_\Omega \phi(\mathcal{N}(\boldsymbol{y}))\mathrm{d}\boldsymbol{x} = \beta \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \sum_{m=1}^{6} \int_{\Omega_{i,j,k,m}} \phi(\mathcal{N}(\boldsymbol{y}))\mathrm{d}\boldsymbol{x}, \tag{31}$$

where $\Omega_{i,j,k,m}$ represents a tetrahedron.

Let $\mathbf{L}^{i,j,k,m}(\boldsymbol{x}) = (L_1^{i,j,k,m}(\boldsymbol{x}), L_2^{i,j,k,m}(\boldsymbol{x}), L_3^{i,j,k,m}(\boldsymbol{x}))$ be the linear interpolation for $\boldsymbol{y}$ in the $\Omega_{i,j,k,m}$, where

$$\begin{aligned} L_1^{i,j,k,m}(\boldsymbol{x}) &= a_1^{i,j,k,m}x_1 + a_2^{i,j,k,m}x_2 + a_3^{i,j,k,m}x_3 + b_1^{i,j,k,m},\\ L_2^{i,j,k,m}(\boldsymbol{x}) &= a_4^{i,j,k,m}x_1 + a_5^{i,j,k,m}x_2 + a_6^{i,j,k,m}x_3 + b_2^{i,j,k,m},\\ L_3^{i,j,k,m}(\boldsymbol{x}) &= a_7^{i,j,k,m}x_1 + a_8^{i,j,k,m}x_2 + a_9^{i,j,k,m}x_3 + b_3^{i,j,k,m}. \end{aligned} \tag{32}$$

Then, in each tetrahedron $\Omega_{i,j,k,m}$, we have $|\nabla \boldsymbol{y}|_F^2 \approx \sum_{l=1}^{9}(a_l^{i,j,k,m})^2$ and

$$\begin{array}{rcrcr} \partial_{x_1}L_1^{i,j,k,m} = a_1^{i,j,k,m}, & & \partial_{x_1}L_2^{i,j,k,m} = a_4^{i,j,k,m}, & & \partial_{x_1}L_3^{i,j,k,m} = a_7^{i,j,k,m},\\ \partial_{x_2}L_1^{i,j,k,m} = a_2^{i,j,k,m}, & & \partial_{x_2}L_2^{i,j,k,m} = a_5^{i,j,k,m}, & & \partial_{x_2}L_3^{i,j,k,m} = a_8^{i,j,k,m},\\ \partial_{x_3}L_1^{i,j,k,m} = a_3^{i,j,k,m}, & & \partial_{x_3}L_2^{i,j,k,m} = a_6^{i,j,k,m}, & & \partial_{x_3}L_3^{i,j,k,m} = a_9^{i,j,k,m},\\ \det \nabla \boldsymbol{y} \approx & & a_1^{i,j,k,m}a_5^{i,j,k,m}a_9^{i,j,k,m} & + & a_2^{i,j,k,m}a_6^{i,j,k,m}a_7^{i,j,k,m}\\ & & +a_4^{i,j,k,m}a_8^{i,j,k,m}a_3^{i,j,k,m} & - & a_2^{i,j,k,m}a_4^{i,j,k,m}a_9^{i,j,k,m}\\ & & -a_1^{i,j,k,m}a_6^{i,j,k,m}a_8^{i,j,k,m} & - & a_3^{i,j,k,m}a_5^{i,j,k,m}a_7^{i,j,k,m}. \end{array} \tag{33}$$

Here, we construct $D_l, 1 \le l \le 9$:

$$\begin{array}{lll} D_1 = [M_1, 0, 0], & D_4 = [0, M_1, 0], & D_7 = [0, 0, M_1],\\ D_2 = [M_2, 0, 0], & D_5 = [0, M_2, 0], & D_8 = [0, 0, M_2],\\ D_3 = [M_3, 0, 0], & D_6 = [0, M_3, 0], & D_9 = [0, 0, M_3], \end{array} \tag{34}$$

where M_1, M_2, and M_3 are the discrete operators of ∂_{x_1}, ∂_{x_2}, and ∂_{x_3}, respectively, and how to construct them is shown in Appendix C.

Then we denote $D_lY = (a_l^{1,1,1,1}, \ldots, a_l^{n_1,n_2,n_3,6})^T \in \mathbb{R}^{6N}$, $1 \le l \le 9$, and set

$$\begin{aligned} \vec{\boldsymbol{q}}^1(Y) &= \sum_{l=1}^{9} D_lY \odot D_lY,\\ \vec{\boldsymbol{q}}^2(Y) &= D_1Y \odot D_5Y \odot D_9Y + D_2Y \odot D_6Y \odot D_7Y + D_4Y \odot D_8Y \odot D_3Y\\ &\quad - D_2Y \odot D_4Y \odot D_9Y - D_1Y \odot D_6Y \odot D_8Y - D_3Y \odot D_5Y \odot D_7Y,\\ \vec{\boldsymbol{r}}^1(Y) &= (\vec{\boldsymbol{q}}^1(Y))^{1/2} - \sqrt{3}(\vec{\boldsymbol{q}}^2(Y))^{1/3},\\ \vec{\boldsymbol{r}}^2(Y) &= 1./((\vec{\boldsymbol{q}}^1(Y))^{1/2} + \sqrt{3}(\vec{\boldsymbol{q}}^2(Y))^{1/3}),\\ \vec{\boldsymbol{r}}(Y) &= \vec{\boldsymbol{r}}^1(Y) \odot \vec{\boldsymbol{r}}^2(Y), \end{aligned} \tag{35}$$

where $\odot$ denotes the Hadamard product of two matrices and ./ denotes the componentwise division. Then we have the following approximation:

$$\mathcal{S}_{\text{New}}(\boldsymbol{y}) \approx \frac{\beta h}{6}\boldsymbol{\phi}(\vec{\boldsymbol{r}}(Y))e^T, \tag{36}$$

where $\boldsymbol{\phi}(\vec{\boldsymbol{r}}(Y)) = (\phi(\vec{\boldsymbol{r}}(Y)_1), \ldots, \phi(\vec{\boldsymbol{r}}(Y)_{6N}))$ and $e = (1, \ldots, 1) \in \mathbb{R}^{6N}$.

Finally, combining formulae (28), (29), (30), and (36), we get the discretized formulation for (18):

$$\begin{aligned} \min_Y \mathcal{J}(Y) = \frac{h}{2}(\vec{T}(PY) - \vec{R})^T(\vec{T}(PY) - \vec{R}) \ + \frac{\alpha_1 h}{2}(Y - X)^T A^T A(Y - X)\\ + \frac{\alpha_2 h}{2}(Y - X)^T B^T B(Y - X) \ + \frac{\beta h}{6}\boldsymbol{\phi}(\vec{\boldsymbol{r}}(Y))e^T, \end{aligned} \tag{37}$$

where h is as defined in (27).

Remark 3. (i) In the implementation, we impose the Dirichlet boundary condition; namely, $\boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{x}$ when $\boldsymbol{x} \in \partial\Omega$. This is a suitable assumption in image registration, which means that we assume that the transformation is deformed in the interior region. However, if a Neumann boundary condition must be used, we could simply modify our formulation to incorporate the changes at boundaries.

(ii) Since Y does not, in general, correspond to voxel points and the interpolation operator is active at all steps (this is typical of an image registration model), here we choose cubic-spline interpolation [40] to compute $\vec{T}(PY)$. Linear interpolation cannot be applied because it is not differentiable at grid points.

In image registration, the number of variables is usually huge, and the dimension of the resulting optimization problem is also huge. For example, when the size of the given images is $128 \times 64 \times 128$, the number of unknowns is over 3 million ($3 \times 129 \times 65 \times 129$). Hence, designing an efficient and converging solver is of crucial importance.

A search method. The iterative scheme for solving an unconstrained optimization problem is

$$Y^{k+1} = Y^k + \theta^k \delta Y^k, \tag{38}$$

where Y^k is the current iterative point, Y^{k+1} is the next iterative point, θ^k is the step length obtained by an Armijo strategy, and δY^k is the search direction. Here, for finding the step length θ^k, the Armijo strategy with backtracking is [30] is crucial for energy reduction along a descent direction. However, the equation $H^k \delta Y^k = -d^k_{\mathcal{J}}$ with the exact Hessian H^k of (37) is not feasible due to lack of definiteness (here $d^k_{\mathcal{J}}$ is the gradient of $\mathcal{J}$ at Y^k), and so guaranteeing that δY^k is a descent direction is ensured by our choice of $\hat{H}^k$ (approximating H^k). In the numerical implementation, we choose a Gauss–Newton algorithm with a line search method to solve the resulting unconstrained optimization problems (37).

Here we propose a generalized Gauss–Newton direction from solving the generalized Gauss–Newton system:

$$\hat{H}^k \delta Y^k = -d^k_{\mathcal{J}}, \tag{39}$$

where $\hat{H}^k$ is the generalized Gauss–Newton matrix of $\mathcal{J}$ at Y^k. The key message is that the generalized Gauss–Newton matrix $\hat{H}^k$ is some positive definite matrix, approximating the full Hessian matrix H^k of (37) (since H^k is not symmetric positive definite). Construction of a suitable $\hat{H}^k$ is a key step.

Below we shall examine and approximate the three constituents of the exact Hessian separately.

First, for the discretized SSD (28), its gradient and Hessian are, respectively,

$$\left\{ \begin{array}{lll} d_1 & = & hP^T \vec{T}^T_{\tilde{\mathbf{Y}}} (\vec{T}(\tilde{\mathbf{Y}}) - \vec{R}), \\ H_1 & = & hP^T (\vec{T}^T_{\tilde{\mathbf{Y}}} \vec{T}_{\tilde{\mathbf{Y}}} + \sum_{l=1}^N (\vec{T}(\tilde{\mathbf{Y}}) - \vec{R})_l \nabla^2 (\vec{T}(\tilde{\mathbf{Y}}) - \vec{R})_l) P, \end{array} \right. \tag{40}$$

where $\tilde{\mathbf{Y}} = PY$ and $\vec{T}_{\tilde{\mathbf{Y}}} = \frac{\partial \vec{T}(\tilde{\mathbf{Y}})}{\partial \tilde{\mathbf{Y}}}$ denotes the Jacobian of $\vec{T}$ with respect to $\tilde{\mathbf{Y}}$. Since we cannot guarantee that H_1 is positive semidefinite, here, we omit the second order term to obtain the approximated Hessian of (28):

$$\hat{H}_1 = hP^T(\vec{T}_{\tilde{\mathbf{Y}}}^T\vec{T}_{\tilde{\mathbf{Y}}})P, \tag{41}$$

which is positive semidefinite.

Second, for the discretized first and second order regularizers (29) and (30), the gradient and Hessian are in the following:

$$\left\{\begin{array}{lcl} d_2 & = & (\alpha_1 hA^TA + \alpha_2 hB^TB)(Y - X), \\ H_2 & = & \alpha_1 hA^TA + \alpha_2 hB^TB. \end{array}\right. \tag{42}$$

Finally, for the discretized new regularizer (36), the gradient and Hessian are as follows:

$$\left\{\begin{array}{lcl} d_3 & = & \frac{\beta h}{6}\mathrm{d}\vec{\boldsymbol{r}}^T\mathrm{d}\boldsymbol{\phi}(\vec{\boldsymbol{r}}), \\ H_3 & = & \frac{\beta h}{6}(\mathrm{d}\vec{\boldsymbol{r}}^T\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\mathrm{d}\vec{\boldsymbol{r}} + \sum_{l=1}^{6N}[\mathrm{d}\boldsymbol{\phi}(\vec{\boldsymbol{r}})]_l\nabla^2(\vec{\boldsymbol{r}})_l), \end{array}\right. \tag{43}$$

where $\mathrm{d}\boldsymbol{\phi}(\vec{\boldsymbol{r}}) = (\phi'((\vec{\boldsymbol{r}})_1), \dots, \phi'((\vec{\boldsymbol{r}})_{6N}))^T$ is the vector of derivatives of $\boldsymbol{\phi}$ at all tetrahedrons,

$$\begin{array}{lcl} \mathrm{d}\vec{\boldsymbol{r}} & = & \mathrm{diag}(\vec{\boldsymbol{r}}^1)\mathrm{d}\vec{\boldsymbol{r}}^2 + \mathrm{diag}(\vec{\boldsymbol{r}}^2)\mathrm{d}\vec{\boldsymbol{r}}^1, \\ \mathrm{d}\vec{\boldsymbol{r}}^1 & = & \frac{1}{2}\,\mathrm{diag}(1./(\vec{\boldsymbol{q}}^1)^{\frac{1}{2}})\mathrm{d}\vec{\boldsymbol{q}}^1 - \frac{\sqrt{3}}{3}\,\mathrm{diag}(1./(\vec{\boldsymbol{q}}^2)^{\frac{2}{3}})\mathrm{d}\vec{\boldsymbol{q}}^2, \\ \mathrm{d}\vec{\boldsymbol{r}}^2 & = & -\,\mathrm{diag}(\vec{\boldsymbol{r}}^2 \odot \vec{\boldsymbol{r}}^2)[\frac{1}{2}\,\mathrm{diag}(1./(\vec{\boldsymbol{q}}^1)^{\frac{1}{2}})\mathrm{d}\vec{\boldsymbol{q}}^1 + \frac{\sqrt{3}}{3}\,\mathrm{diag}(1./(\vec{\boldsymbol{q}}^2)^{\frac{2}{3}})\mathrm{d}\vec{\boldsymbol{q}}^2], \\ \mathrm{d}\vec{\boldsymbol{q}}^1 & = & 2\sum_{l=1}^{9}\mathrm{diag}(D_lY)D_l, \\ \mathrm{d}\vec{\boldsymbol{q}}^2 & = & \mathrm{diag}(D_5Y \odot D_9Y - D_6Y \odot D_8Y)D_1 \\ & + & \mathrm{diag}(D_6Y \odot D_7Y - D_4Y \odot D_9Y)D_2 \\ & + & \mathrm{diag}(D_4Y \odot D_8Y - D_5Y \odot D_7Y)D_3 \\ & + & \mathrm{diag}(D_8Y \odot D_3Y - D_2Y \odot D_9Y)D_4 \\ & + & \mathrm{diag}(D_1Y \odot D_9Y - D_3Y \odot D_7Y)D_5 \\ & + & \mathrm{diag}(D_2Y \odot D_7Y - D_1Y \odot D_8Y)D_6 \\ & + & \mathrm{diag}(D_2Y \odot D_6Y - D_3Y \odot D_5Y)D_7 \\ & + & \mathrm{diag}(D_4Y \odot D_3Y - D_1Y \odot D_6Y)D_8 \\ & + & \mathrm{diag}(D_1Y \odot D_5Y - D_2Y \odot D_4Y)D_9, \end{array} \tag{44}$$

$\mathrm{d}\vec{\boldsymbol{r}}, \mathrm{d}\vec{\boldsymbol{r}}^1, \mathrm{d}\vec{\boldsymbol{r}}^2, \mathrm{d}\vec{\boldsymbol{q}}^1, \mathrm{d}\vec{\boldsymbol{q}}^2$ are the Jacobian of $\vec{\boldsymbol{r}}, \vec{\boldsymbol{r}}^1, \vec{\boldsymbol{r}}^2, \vec{\boldsymbol{q}}^1, \vec{\boldsymbol{q}}^2$ with respect to Y, respectively, $\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})$ is the Hessian of $\boldsymbol{\phi}$ with respect to $\vec{\boldsymbol{r}}$, which is a diagonal matrix whose ith diagonal element is $\phi''((\vec{\boldsymbol{r}})_i)$, $1 \le i \le 6N$. Here, $\mathrm{diag}(v)$ is a diagonal matrix with v on its main diagonal.

To extract a positive semidefinite part, we again omit the second order term and obtain the following approximated Hessian:

$$\hat{H}_3 = \frac{\beta h}{6}\mathrm{d}\vec{\boldsymbol{r}}^T\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\mathrm{d}\vec{\boldsymbol{r}}. \tag{45}$$

So the generalized Gauss–Newton system is

$$\hat{H}\delta Y = -d_{\mathcal{J}}, \tag{46}$$

where $\hat{H} = \hat{H}_1 + H_2 + \hat{H}_3$ and $d_{\mathcal{J}} = d_1 + d_2 + d_3$.

Remark 4. Here, by construction, $\hat{H}$ is indeed a positive definite matrix since H_2 is positive definite under the Dirichlet boundary conditions and $\hat{H}_1$ and $\hat{H}_3$ are both positive semidefinite.

The overall numerical solution scheme is summarized in Algorithm 1 below. Here, we choose the stopping criteria consistent with the literature [40, 50]; namely, when the changes in the objective function, the norm of the update, and the norm of the gradient are all sufficiently small, the iterations are terminated. In each iteration, we need to solve the generalized Gauss–Newton system (46) to find the search direction δU. Here, we choose MINRES [3, 42] to solve this system, and, in the implementation, the tolerance for the relative residual is set to 0.1. Except for the diagonal preconditioner, we also consider a preconditioner L, which is composed of the diagonals of blocks of the approximated Hessian $\hat{H}$ shown in Figure 3(b). This choice is motivated by two aspects. On one hand, since the discretized optimization problem is usually large scale and the storage is limited, we cannot explicitly formulate the approximated Hessian $\hat{H}$. In the implementation, the approximated Hessian $\hat{H}$ is stored implicitly, and we just provide a matrix-free version to compute the matrix-vector product $\hat{H}v$. In this way, it is easy to extract the diagonals of the blocks of the approximated Hessian $\hat{H}$. One the other hand, after a permutation E, the preconditioner L can be converted into ELE, a diagonal matrix of blocks shown in Figure 3(c). Hence, solving $Lx = b$ in each iteration is very fast. The efficiency of this preconditioner L is also illustrated in section 5 numerically. In Appendices D and E, we give the details of how to compute the matrix-vector product $\hat{H}v$, the diagonal of $\hat{H}$, and the preconditioner L.

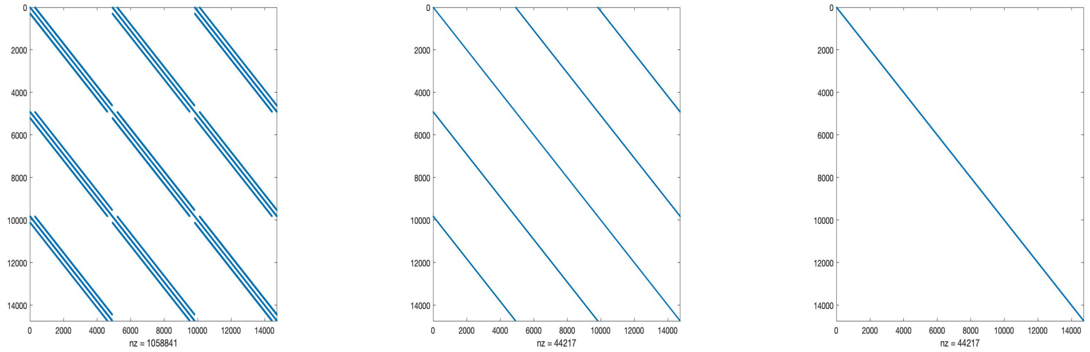
Algorithm 1. Generalized Gauss–Newton scheme by using Armijo line search for image registration: $Y \leftarrow \text{GNAIR}(\alpha_1, \alpha_2, \beta, Y^0, T, R)$.

```
Step 1: Given $Y^0$;
Step 2: For (37), compute $\mathcal{J}(Y^0)$, $d^0_{\mathcal{J}}$ and $\hat{H}^0$;
Step 3: Set $k = 0$;
while "the stopping criteria are not satisfied" do
    — Solve $\hat{H}^k \delta Y^k = -d^k_{\mathcal{J}}$ from (46);
    — Update $Y^{k+1}$ by an Armijo step via (38);
    — $k = k + 1$;
    — compute $\mathcal{J}(Y^k)$, $d^k_{\mathcal{J}}$ and $\hat{H}^k$;
end while
```

Remark 5. Here the generalized Gauss–Newton system is symmetric positive definite. Apart from preconditioning, we choose MINRES rather than the standard conjugate gradient method (CG); this is based on experimental performance where we find that MINRES is faster, leading to a more accurate transformation than CG under the same stopping criteria. Alternatively we could use a restarted GMRES method as the inner solver.

For Algorithm 1, we have the following global convergence result.

Theorem 8. *Let T and R be twice continuously differentiable. For* (37)*, if choosing a sufficiently large β and setting the discretized identity map as Y^0, then each iterate Y^k generated*



(a) $\hat{H}$ (b) L (c) ELE

Figure 3. *The structure of the approximated Hessian $\hat{H}$ (left), the structure of the preconditioner L (middle), and the structure of the preconditioner L after a permutation E (right). L is composed of the diagonals of blocks of the approximated Hessian $\hat{H}$. The size of the matrix is* 14739 $(3 \times 17 \times 17 \times 17)$.

by Algorithm 1 is in $\mathcal{Y}$ for some small constant ϵ:

$$\mathcal{Y} = \{Y \mid (\vec{\boldsymbol{r}}(Y))_l \leq 1-\epsilon, 1 \leq l \leq 6N\}. \tag{47}$$

In addition, we have

$$\lim_{k\to\infty} d_{\mathcal{J}}(Y^k) = 0, \tag{48}$$

and hence any limit point of the sequence of iterates produced by Algorithm 1 is a stationary point Y^* in $\mathcal{Y}$. The stationary point Y^* is also a discretized one-to-one transformation.

Proof. Consider the following space:

$$\overline{\mathcal{Y}} = \{Y \mid (\vec{\boldsymbol{r}}(Y))_l < 1, 1 \leq l \leq 6N\}. \tag{49}$$

If the kth iteration Y^k is in $\overline{\mathcal{Y}}$, then by [10, Lemma 1], the Armijo line search can give the $k+1$th iteration Y^{k+1} that is also in $\overline{\mathcal{Y}}$. Since, in the implementation, the initial iteration Y^0 is the discretized identity map and we have $\vec{\boldsymbol{r}}(Y^0) = \mathbf{0}$, then by Algorithm 1 it can generate a sequence $(Y^k)_{k\in\mathbb{N}}$, which is in the space $\overline{\mathcal{Y}}$. Furthermore, by the sufficient decrease condition in the Armijo line search, we have

$$\mathcal{J}(Y^0) > \mathcal{J}(Y^1) > \cdots > \mathcal{J}(Y^k) > \cdots. \tag{50}$$

Then with a sufficiently large β, we can ensure that the generated sequence $(Y^k)_{k\in\mathbb{N}}$ is in the following space:

$$\mathcal{Y} = \{Y \mid (\vec{\boldsymbol{r}}(Y))_l \leq 1-\epsilon, 1 \leq l \leq 6N \text{ for some small } \epsilon\}. \tag{51}$$

Since the Dirichlet boundary condition is applied, $\|Y\|$ is bounded and $\vec{\boldsymbol{r}}(Y)$ is a continuous mapping from a compact set to $\mathbb{R}^{6N}$. Hence, for some small $\epsilon > 0$, $\mathcal{Y}$ is compact.

Next, we need to verify that the following conditions are satisfied:

(1) $d_{\mathcal{J}}$ is Lipschitz continuous.
(2) For all k, $\hat{H}^k$ is symmetric and positive definite.
(3) There exist constant $\overline{\kappa}$ and ζ such that the condition number $\kappa(\hat{H}^k) \leq \overline{\kappa}$ and the norm $\|\hat{H}^k\| \leq \zeta$ for all k.
(4) $\mathcal{J}(Y)$ has a lower bound.

Because T and R are twice continuously differentiable, (37) is twice continuously differentiable with respect to $Y \in \mathcal{Y}$ and $d_{\mathcal{J}}$ is Lipschitz continuous.

We have remarked that $\hat{H}^k$ is symmetric positive definite by construction. In each iteration, $H_2^k = \alpha_1 h A^T A + \alpha_2 h B^T B$ is constant, and we can set $\|H_2^k\| = \zeta_2$. For $\hat{H}_1^k = hP^T(\vec{T}^T_{\tilde{\mathbf{Y}}}\vec{T}_{\tilde{\mathbf{Y}}})P$, we get its upper bound ζ_1 because T is twice continuously differentiable and $\mathcal{Y}$ is compact. In addition, ϕ is twice continuously differentiable; then we have $\|\hat{H}_3^k\| \leq \frac{\beta h}{6}\|\mathrm{d}\vec{\boldsymbol{r}}^T\|\|\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\|\|\mathrm{d}\vec{\boldsymbol{r}}\| \leq \zeta_3$. Hence, we have

$$\|\hat{H}^k\| \leq \|\hat{H}_1^k\| + \|H_2^k\| + \|\hat{H}_3^k\| \leq \zeta_1 + \zeta_2 + \zeta_3 = \zeta. \tag{52}$$

Let σ be the minimum eigenvalue of H_2^k. Then the smallest eigenvalue $\lambda_{\min}$ of $\hat{H}^k$ should be larger than σ. Due to $\lambda_{\max} \leq \|\hat{H}^k\|$, the largest eigenvalue $\lambda_{\max}$ of $\hat{H}^k$ should be smaller than ζ. So set $\overline{\kappa} = \frac{\zeta}{\sigma}$, and the condition number of $\hat{H}^k$ is smaller than $\overline{\kappa}$.

Finally, we can see that a lower bound of (37) is 0 since it is nonnegative. Since the above listed four conditions are satisfied, according to [30, Thm. 3.2.4], we complete the proof. ■

In the above result, we assume that the initial guess Y^0 is an identity (or equally the deformation field is zero). However, though such a zero start is enough for convergence, it is a common practice to adopt a multilevel strategy to obtain a better initial guess and speed up image registration. Specifically, we first coarsen the template and the reference by $L \geq 1$ levels recursively and then, starting on the coarsest level, register the coarsened images before interpolating to the next finer level until we are back to the finest level. There are two issues to consider: (i) L should be such that the images on the coarsest level still possess the large differences in the pair of images in order for registration to be meaningful. (ii) Coarse to fine level interpolation should ensure that the interpolation Y^0 still remains in space $\overline{\mathcal{Y}}$ (or diffeomorphic) on that level. The most important advantage of the multilevel strategy is that it can use less time to provide a good initial guess because there are fewer variables on coarser levels than on the fine level.

As mentioned earlier, an alternative to a Gauss–Newton method is ADMM, where one splits an original problem into several subproblems. For the proposed model (18), we introduce one auxiliary variable $\boldsymbol{v}$ and have the following equivalent formulation:

$$\begin{aligned}\min_{\boldsymbol{y},\boldsymbol{v}} \frac{1}{2}\int_\Omega (T\circ\boldsymbol{y} - R)^2\mathrm{d}\boldsymbol{x} &+ \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y}-\boldsymbol{x})\|_F^2\mathrm{d}\boldsymbol{x} + \frac{\alpha_2}{2}\int_\Omega \|\nabla^2(\boldsymbol{y}-\boldsymbol{x})\|_F^2\mathrm{d}\boldsymbol{x}\\ &+ \beta\int_\Omega \phi\left(\frac{\|\boldsymbol{v}\|_F - \sqrt{3}(\det\boldsymbol{v})^{1/3}}{\|\boldsymbol{v}\|_F + \sqrt{3}(\det\boldsymbol{v})^{1/3}}\right)\mathrm{d}\boldsymbol{x} \qquad \text{s.t. } \boldsymbol{v} = \nabla\boldsymbol{y}.\end{aligned} \tag{53}$$

After discretization, we get the following constrained optimization problem:

$$\min_{Y,V} \frac{h}{2}(\vec{T}(PY)-\vec{R})^T(\vec{T}(PY)-\vec{R}) + \frac{\alpha_1 h}{2}(Y-X)^T A^T A(Y-X) + \frac{\alpha_2 h}{2}(Y-X)^T B^T B(Y-X) + \frac{\beta h}{6}\boldsymbol{\phi}(\vec{\boldsymbol{s}}(V))e^T \qquad \text{s.t. } V = DY, \tag{54}$$

where $D = [D_1^T, \ldots, D_9^T]^T$ is the first order discrete operator based on (34) and $\vec{\boldsymbol{s}}$ is just defined following the definition of $\vec{\boldsymbol{r}}$ in (35).

To investigate the convergence of ADMM for (54), we first review a recent convergence result.

Theorem 9 (see [49]). *Consider the following problem:*

$$\min_{x,y} g(x) + h(y) \qquad s.t.\ Px + Qy = 0, \tag{55}$$

where $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, $h : \mathbb{R}^q \to \mathbb{R}$, $P \in \mathbb{R}^{m\times n}$, and $Q \in \mathbb{R}^{m\times q}$. Assume the following hold:

- A1 *(coercivity) Define the feasible set $\mathcal{F} := \{(x,y) \in \mathbb{R}^{n+q} : Px + Qy = 0\}$. The objective function $g(x) + h(y)$ is coercive over this set, that is, $g(x) + h(y) \to \infty$ if $(x,y) \in \mathcal{F}$ and $\|x,y\| \to \infty$.*
  The assumption A1 *can be dropped if the feasible set of (x,y) is bounded.*
- A2 *(feasibility) Im$(P) \subseteq$ Im(Q), where Im$(\cdot)$ returns the image of a matrix.*
- A3 *(Lipschitz subminimization paths)*
  - (a) $\operatorname{argmin}_y\{h(y) : Qy = u\}$ *has a unique minimizer. $H : \mathrm{Im}(Q) \to \mathbb{R}^q$ defined by $H(u) := \operatorname{argmin}_y\{h(y) : Qy = u\}$ is a Lipschitz continuous map.*
  - (b) $\operatorname{argmin}_x\{g(x) : Px = u\}$ *has a unique minimizer. $F : \mathrm{Im}(P) \to \mathbb{R}^n$ defined by $F(u) := \operatorname{argmin}_x\{g(x) : Px = u\}$ is a Lipschitz continuous map.*
- A4 *(objective-f regularity) g is Lipschitz differentiable with constant L_f.*
- A5 *(objective-h regularity) h is Lipschitz differentiable with constant L_g.*

Then ADMM converges subsequently for any sufficient large penalty parameter, that is, starting from any initial guess point, it generates a sequence that is bounded, has at least one limit point, and each limit point is a stationary point of its augmented Lagrangian function.

To apply Theorem 9, we convert (54) into the following form:

$$\min_{Y,V} g(Y) + h(V) \qquad \text{s.t. } DY - V = 0, \tag{56}$$

where $g(Y) = \frac{h}{2}(\vec{T}(PY)-\vec{R})^T(\vec{T}(PY)-\vec{R}) + \frac{\alpha_1 h}{2}(Y-X)^T A^T A(Y-X) + \frac{\alpha_2 h}{2}(Y-X)^T B^T B(Y-X)$ and $h(V) = \frac{\beta h}{6}\boldsymbol{\phi}(\vec{\boldsymbol{s}}(V))e^T$.

For (56), the feasible set (Y,V) is bounded, and then A1 can be dropped, due to the Dirichlet boundary conditions. A2 and A3(a) are trivial because for (56), $Q = -I$. Again, by imposing the Dirichlet boundary conditions, $P = D$ is full column rank and then A3(b) holds. Here, since we assume that T and R are twice continuously differentiable and the feasible set is bounded, A4 holds. However, since some components of $\vec{\boldsymbol{s}}(V)$ may be infinity, the gradient of $h(V)$ can be infinity and is not Lipschitz continuous. Hence A5 does not hold, and consequently Theorem 9 cannot be applied.

Alternatively, we could introduce two auxiliary variables $\boldsymbol{v}$ and $\boldsymbol{w}$ to build a three-block ADMM:

(57)
$$\min_{\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \frac{1}{2}\int_\Omega (T\circ\boldsymbol{y}-R)^2\mathrm{d}\boldsymbol{x} + \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y}-\boldsymbol{x})\|_F^2\mathrm{d}\boldsymbol{x} + \frac{\alpha_2}{2}\int_\Omega \|\nabla^2(\boldsymbol{y}-\boldsymbol{x})\|_F^2\mathrm{d}\boldsymbol{x} + \beta\int_\Omega \phi(\boldsymbol{w})\mathrm{d}\boldsymbol{x}$$
$$\text{s.t. } \boldsymbol{v}=\nabla\boldsymbol{y},\ \ \boldsymbol{w}=\frac{\|\boldsymbol{v}\|_F-\sqrt{3}(\det\boldsymbol{v})^{1/3}}{\|\boldsymbol{v}\|_F+\sqrt{3}(\det\boldsymbol{v})^{1/3}}.$$

Clearly each resulting subproblem is more easily solved than for (54). However, since (57) contains a nonlinear constraint, the convergence of ADMM still cannot be established. This can be one future research direction.

4. Other possible regularizers.

In this section, we give another two possible 3D regularizers for the orientation-preserving image registration.

First, making use of (10) from the LLL work and extending it beyond landmark registration, we could consider the following regularizer for a 3D map $\boldsymbol{f}$.

Definition 10. *If the map* $\boldsymbol{f}(x_1,x_2,x_3)=(y_1(x_1,x_2,x_3),y_2(x_1,x_2,x_3),y_3(x_1,x_2,x_3))$ *is continuously differentiable, then we define*

$$\mathcal{N}_1(\boldsymbol{f})=\frac{1}{3}\left(\frac{\|\nabla\boldsymbol{f}\|_F^2}{(\det\nabla\boldsymbol{f})^{2/3}}\right) \tag{58}$$

as a new regularizer for a 3*D map* $\boldsymbol{f}$.

Then, the following lemma shows some properties of (58).

Lemma 11. *Regularizer* $\mathcal{N}_1$ *from* (58) *possesses the following properties:*
P1 $\mathcal{N}_1(\boldsymbol{f})=1 \Leftrightarrow$ *the singular values of* $\nabla\boldsymbol{f}$ *are equal;*
P2 $1\le\mathcal{N}_1(\boldsymbol{f})\le\infty$;
P3 $\mathcal{N}_1(\boldsymbol{f})=\infty\Leftrightarrow\det\nabla\boldsymbol{f}=0$.

Unfortunately $\mathcal{N}_1$ does not share all the properties of $\mathcal{N}$.

Second, we consider another possible regularizer. Since (58) represents a dilatation in 3D according to the LLL work (see (7)), we may use $\mathcal{N}_1$ to define the distortion in 3D by

$$\mathcal{N}_2(\boldsymbol{f})=\frac{\mathcal{N}_1(\boldsymbol{f})-1}{\mathcal{N}_1(\boldsymbol{f})+1}. \tag{59}$$

Then, we can rewrite the above to define another new regularizer.

Definition 12. *If the map* $\boldsymbol{f}(x_1,x_2,x_3)=(y_1(x_1,x_2,x_3),y_2(x_1,x_2,x_3),y_3(x_1,x_2,x_3))$ *is continuously differentiable, then we define*

$$\mathcal{N}_2(\boldsymbol{f})=\frac{\|\nabla\boldsymbol{f}\|_F^2-3(\det\nabla\boldsymbol{f})^{2/3}}{\|\nabla\boldsymbol{f}\|_F^2+3(\det\nabla\boldsymbol{f})^{2/3}} \tag{60}$$

as a new regularizer for a 3*D map* $\boldsymbol{f}$.

Similarly, we can show that $\mathcal{N}_2(\boldsymbol{f})$ has the following properties.

Lemma 13. *Regularizer $\mathcal{N}_2$ from (60) possesses the following properties:*
P1 $\mathcal{N}_2(\boldsymbol{f}) = 0 \Leftrightarrow$ *the singular values of* $\nabla \boldsymbol{f}$ *are equal;*
P2 $0 \le \mathcal{N}_2(\boldsymbol{f}) \le 1$*;*
P3 $0 \le \mathcal{N}_2(\boldsymbol{f}) < 1 \Leftrightarrow \det \nabla \boldsymbol{f} \neq 0$*;*
P4 $\mathcal{N}_2(\boldsymbol{f}) = 1 \Leftrightarrow \det \nabla \boldsymbol{f} = 0$*.*

Clearly $\mathcal{N}_2$ seems better than $\mathcal{N}_1$ in that it shares more properties of $\mathcal{N}$.

Therefore, based on $\mathcal{N}_1, \mathcal{N}_2$, we can present two respective models as follows:

$$\min_{\boldsymbol{y}} \frac{1}{2}\int_\Omega (T\circ \boldsymbol{y} - R)^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y}-\boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_2}{2}\int_\Omega \|\nabla^2(\boldsymbol{y}-\boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \beta\int_\Omega \phi(\mathcal{N}_2(\boldsymbol{y}))\mathrm{d}\boldsymbol{x}, \tag{61}$$

where $\phi(v) = v^2/((v-1)^2+1)$ is the same as what was used in the proposed model (18), and

$$\min_{\boldsymbol{y}} \frac{1}{2}\int_\Omega (T\circ \boldsymbol{y} - R)^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_1}{2}\int_\Omega \|\nabla(\boldsymbol{y}-\boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \frac{\alpha_2}{2}\int_\Omega \|\nabla^2(\boldsymbol{y}-\boldsymbol{x})\|_F^2 \mathrm{d}\boldsymbol{x} + \beta\int_\Omega \phi(\mathcal{N}_1(\boldsymbol{y}))\mathrm{d}\boldsymbol{x}, \tag{62}$$

where $\phi(v) = v^2$ because we promote $\mathcal{N}_1 < \infty$. Here, (62) can be considered as a reasonable modified LLL model under our framework, which is mainly used to make a comparison in the later test since the $\mathcal{N}_1$ comes from the LLL model.

Clearly, P2 from Lemma 11 and P3 from Lemma 13 show that if we just control $\mathcal{N}_1 < \infty$ and $\mathcal{N}_2 < 1$, it is not sufficient to ensure that the obtained transformation is orientation-preserving, since the term $(\det \nabla \boldsymbol{f})^{2/3}$ is never negative. Adding an explicit constraint such as $\det \nabla \boldsymbol{y} > 0$ defeats the idea of unconstrained optimization. Hence, it remains a problem to modify these two models (61) and (62) to achieve orientation-preserving transformations. However, a practical strategy has to be using large parameters for $\alpha_1, \alpha_2, \beta$ to balance accuracy and mesh quality (towards quality).

5. Numerical experiments.

In this section, we demonstrate the performance of our new model (18) by three 3D examples. Specifically we shall compare these models:

- NEW from (18)—the proposed new model;
- O1 from (61)—the first alternative model (generalization of LLL);
- O2 LLL from (62)—the second alternative model (generalization of LLL);
- Hyper from (3)—the hyperplastic model which assumes $\det \nabla \boldsymbol{y} \approx 1$;
- LDDMM from (5)—the LDDMM model.

All the numerical experiments are run in MATLAB 2019a on a MacBook Pro with 2.2 GHz Quad-Core Intel Core i7 microprocessor and 16 GB of memory. As a comparison, we compare our model (18) with the state-of-the-art methods, the hyperelastic model (Hyper [8]), LDDMM [36], and the modified LLL model (O2). The codes of Hyper and LDDMM are based on FAIR [40], which can be downloaded from https://github.com/C4IR/FAIR.m. The implementation of O1 and O2 are similar to Algorithm 1, including discretization and deriving the approximated Hessian. To measure the quality of the registration, we consider the following quantities:

- Relative SSD (Re_SSD) defined by $\frac{\|T\circ \boldsymbol{y}-R\|^2}{\|T-R\|^2}$ to measure the relative residual.
- The minimum of the Jacobian determinant of the transformation ($\min \det \nabla \boldsymbol{y}$) and the maximum of the Jacobian determinant of the transformation ($\max \det \nabla \boldsymbol{y}$) to

measure the quality of the mesh.

- Dice similarity coefficient (DSC) defined by $\frac{2|\Omega_{T\circ\boldsymbol{y}}\cap\Omega_R|}{|\Omega_{T\circ\boldsymbol{y}}|+|\Omega_R|}$ to evaluate the similarity of the volume. Here, $\Omega_{T\circ\boldsymbol{y}}$ is the relevant volume part of the deformed template, and Ω_R is the relevant volume part of the reference.

5.1. Test 1: Comparison of models for a pair of synthetic images. We construct a synthetic example (a big ball and a small collapsed ball) to highlight the advantage of our model (18) over the other models. A comparison task is a highly nontrivial matter because there are potentially unfair choices to favor a certain method. To remedy this, we tried to use other colleagues' codes whenever possible to reduce bias, and we tuned parameters of other compared methods to show only their most applicable results.

Figure 4 shows the template and the reference. Here, the dimension of the given images is $64 \times 64 \times 64$, and the domain of the images is $[0, 64]^3$. In the implementation, we employ a four-step multilevel strategy for all methods and discretize the images by using regular meshes with $8 \times 8 \times 8$, $16 \times 16 \times 16$, $32 \times 32 \times 32$, and $64 \times 64 \times 64$, respectively. On the finest level, the number of the unknowns in this example is 823875.

For the choice of the parameters of Hyper, we use the (recommended) parameters $\alpha_l = 100$ (length regularizer), $\alpha_s = 1$ (surface regularizer), and $\alpha_v = 10$ (volume regularizer). For LDDMM, a suitable choice is $\alpha = 1200$ to control the smoothness of the velocity and $N_t = 10$ as the number of time steps for computing the characteristic [36]. For NEW, O1, and O2, we fix $\alpha_1 = 100$, $\alpha_2 = 0.1$ and choose $\beta = 6200$, $\beta = 2000$, and $\beta = 50$, respectively.

Figure 5 shows the deformed templates obtained by these five models, and Table 1 gives the corresponding measurements. Using the symbol $>$ to denote "better than," the comparisons may be summarized as follows:

- *Visual differences.* From Figure 5, we can see that NEW, O1, and O2 have all generated visually acceptable deformed templates (similar to the reference), but LDDMM and Hyper have not. That is,

$$\text{NEW, O1, and O2} \quad > \quad \text{Hyper and LDDMM.}$$

- *Error (accuracy) differences.* Column 2 of Table 1 shows the relative residuals of five models to inform accuracies of this example. Clearly Hyper and LDDMM are less satisfactory than all the others. Precisely, we see that

$$\text{NEW} \; > \; \text{O1} \; > \; \text{O2} \; > \; \text{Hyper and LDDMM.}$$

  We remark that $\det \nabla \boldsymbol{y} \approx 1$ does not hold.

- *Bijectivity differences.* Columns 3–4 of Table 1 show the minimum and maximum of the Jacobian determinant of the transformation obtained by each model. Although we only require $\min \det \nabla \boldsymbol{y} > 0$ to ensure an orientation-preserving transformation and in this regard all five models are satisfactory, we can notice that the range of the Jacobian determinant of the transformations obtained by NEW, O1, O2, and LDDMM are larger than Hyper since the latter explicitly aims for 1, which is not a reasonable condition in this example.

- *DSC differences.* Column 7 of Table 1 shows the Dice similarity coefficient of these models. Again, we can see that

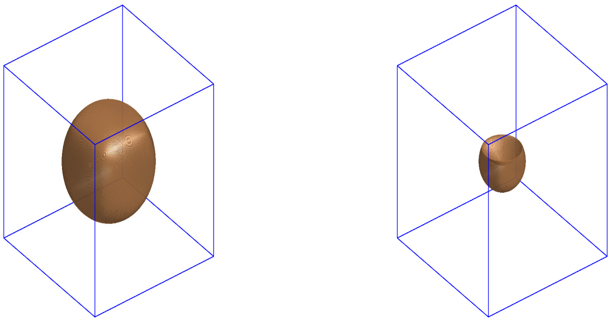
  $$\text{NEW, O1, and O2} \quad > \quad \text{Hyper and LDDMM.}$$

- *Solution speed differences.* Columns 5–6 of Table 1 show the CPU times and iterations of these five models. Clearly we see that

  $$\text{NEW, O1, and O2} \quad > \quad \text{Hyper and LDDMM.}$$

  Here, for LDDMM, since the deformation is large, the main part of its computing time is spent on computing the characteristic of the transport equation accurately.

Hence, for the large deformation problems where volume preservation is not required, our new model NEW can show the advantages over other models.



(a) Template T (b) Reference R

(c) Reference R in axial, coronal, and sagittal views

Figure 4. *Test 1: The first row shows the template and reference. The second row shows the reference in axial, coronal, and sagittal views, respectively. Since the template is a ball, its axial, coronal, and sagittal views are the same.*

5.2. Test of the preconditioner, convergence, and solver. Here, we use Test 1 to investigate the preconditioner, convergence of the algorithm, and the performance of the different solvers for our new models.

Table 1

Test 1: Comparison of the new models with Hyper and LDDMM.

	Re_SSD	$\min\det\nabla\boldsymbol{y}$	$\max\det\nabla\boldsymbol{y}$	DSC	Time (s)	Iter. on each level
NEW	0.08%	0.3583	36.8403	0.9243	15.2	$13, 5, 3, 4$
O1	0.11%	0.3541	37.7971	0.9206	15.5	$10, 4, 3, 4$
O2	0.12%	0.3459	35.4774	0.9227	14.4	$11, 4, 3, 4$
Hyper	1.26%	0.1212	16.4666	0.8746	55.0	$20, 6, 6, 7$
LDDMM	1.31%	0.0001	38.1041	0.8717	797.7	$4, 2, 2, 2$

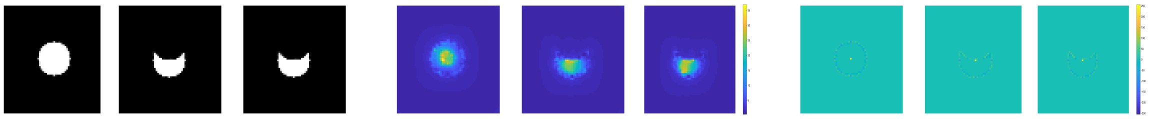
We first investigate the preconditioner mentioned in section 3 for NEW, O1, and O2. From Figure 6, MINRES with the L preconditioner can give the best convergence performance among these solvers. Further, from Table 2, we can still find that MINRES with the L preconditioner uses the least number of iterations and computational time to reach the termination in one iteration of NEW, O1, and O2. Hence, MINRES with the L preconditioner is an effective solver for solving the generalized Gauss–Newton system in the proposed new models.

We next illustrate the convergence of the algorithm for NEW, O1, and O2. Forcing the algorithm to run until the relative norms of the gradients reach 10^{-6} (note that the algorithm can satisfy the stopping criteria in several iterations with a large tolerance, e.g., 10^{-2}), Figure 7 shows the relative norm of the gradient from the first order condition, as shown in Figure 7(a), and the relative energy functional values (Figure 7(b)). We see that the relative norms of the gradient of NEW, O1, and O2 are reduced to 10^{-6}. Clearly, the algorithm for NEW, O1, and O2 is convergent, as predicted by Theorem 8. The convergence is not monotone, which is the usual behavior of an optimization approach for a nonconvex problem [41].

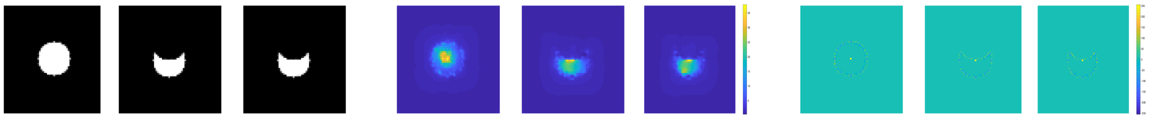
We finally test the performance of different solvers for NEW, O1, and O2. Since the generalized Gauss–Newton system is symmetric positive definite, the conjugate gradient method (CG) seems to be the usual choice. In addition, although GMRES is designed for solving the unsymmetric system, the convergence theory is also suitable for the symmetric system. However, according to Table 3, we can see that the performance of the solver based on GMRES is similar to MINRES, but GMRES spends more running time. In addition, the performance of CG is similar to MINRES for NEW and O2 but worse than MINRES for O1. Especially for MINRES with the L preconditioner, it has the best performance among these different solvers for NEW, O1, and O2, with respect to accuracy and speed. If we apply a strict stopping criterion for CG in O1, from Table 4, their performances can also be comparable with MINRES, but they need more iterations and hence more computational time. Hence, for the key component, solving the generalized Gauss–Newton system in the proposed optimization method, we choose MINRES rather than CG or GMRES.

5.3. Test 2: Comparison of models for a pair of brain images. We illustrate the performance of our model NEW in registering a pair of 3D real-life images. For completeness, we also compare it with the other four models (O1, O2, Hyper, and LDDMM). We choose the human brain images from the data accompanying the software FAIR [40]. The template and the corresponding reference are shown in Figure 8. The size of the given images is $128 \times 64 \times 128$, and the domain of the images is $[0, 20] \times [0, 10] \times [0, 20]$. In the implementation, for all five

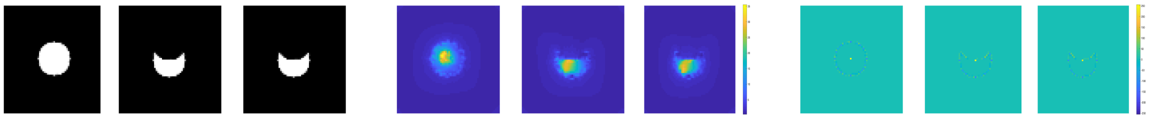
(a) Template (b) Reference (c) Residual before registration



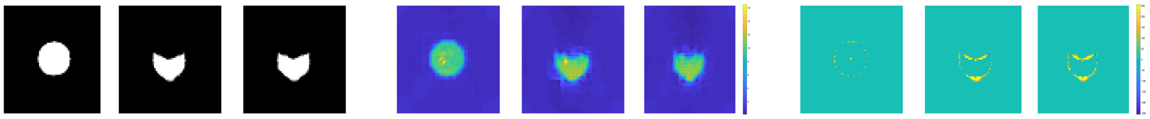
(d) $T(\boldsymbol{y}_{\text{NEW}})$ (e) Relative volume change (f) Residual after registration (0.08%)



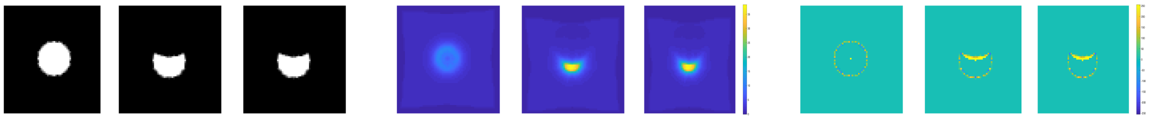
(g) $T(\boldsymbol{y}_{\text{O1}})$ (h) Relative volume change (i) Residual after registration (0.11%)



(j) $T(\boldsymbol{y}_{\text{O2}})$ (k) Relative volume change (l) Residual after registration (0.12%)

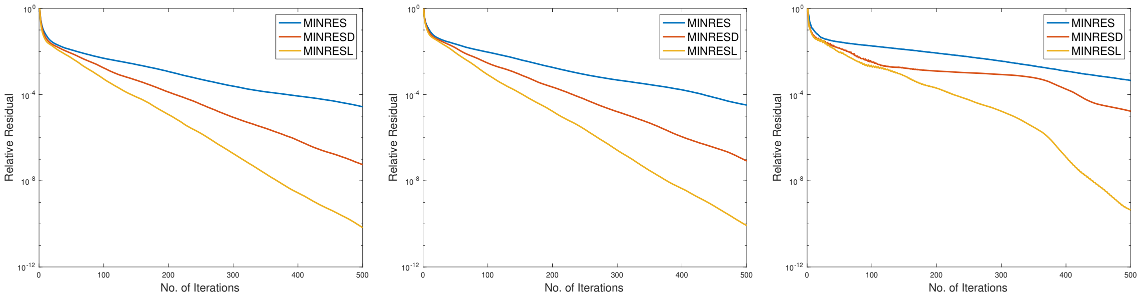


(m) $T(\boldsymbol{y}_{\text{Hyper}})$ (n) Relative volume change (o) Residual after registration (1.26%)



(p) $T(\boldsymbol{y}_{\text{LDDMM}})$ (q) Relative volume change (r) Residual after registration (1.31%)

Figure 5. *The results of Test* 1*: In the first row, there are the template, reference, and the residual before registration in axial, coronal, and sagittal views. The second row to the sixth row show the deformed template, its corresponding relative volume change* ($\det \nabla \boldsymbol{y}$)*, and residual after registration in axial, coronal, and sagittal views obtained by NEW,* O1*,* O2*, Hyper, and LDDMM, respectively. The percentage represents the relative residual.*



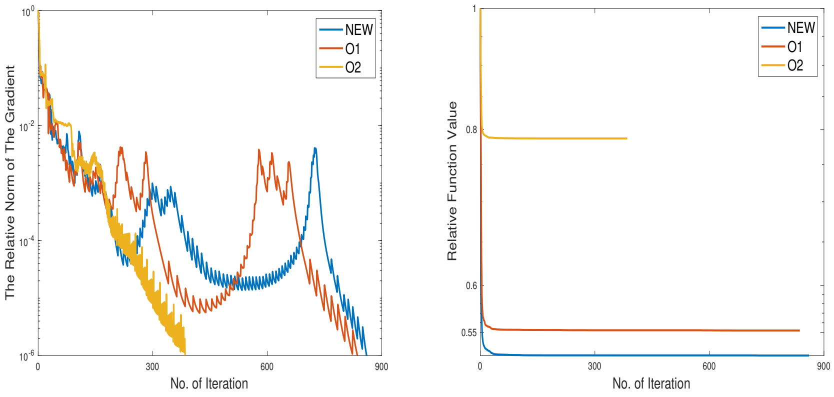
(a) NEW (b) O1 (c) O2

Figure 6. *Residual plots: Performance of different preconditioners for one iteration of NEW,* O1*, and* O2 *in Test* 1*. Here, MINRES, MINRESD, and MINRESL represent that the solver is MINRES without preconditioner, with diagonal preconditioner, and with L preconditioner, respectively.*

Table 2

The number of iterations needed to reach the termination for MINRES with different preconditioners in Test 1*.*

	NEW		O1		O2	
	No. of Iter	Time (s)	No. of Iter	Time (s)	No. of Iter	Time (s)
MINRES	6	1.7	9	2.3	10	2.5
MINRESD	5	1.5	7	1.9	5	1.5
MINRESL	4	1.5	6	1.9	4	1.5



(a) Relative norm of the gradient of NEW, O1, and O2

(b) Relative function values of NEW, O1, and O2

Figure 7. *The relative norm of the gradient and relative function values of NEW,* O1*, and* O2 *in Test* 1*.*

models, we employ a four-step multilevel strategy which is to discretize the images in the following different resolutions: $16 \times 8 \times 16$, $32 \times 16 \times 32$, $64 \times 32 \times 64$, and $128 \times 64 \times 128$. The number of unknowns on the finest level in this example is 3244995.

Here, for the parameters of Hyper, we choose the default parameters provided by FAIR [40], $\alpha_l = 100$, $\alpha_s = 10$, and $\alpha_v = 100$. For LDDMM, we set $\alpha = 200$ to control the smoothness of the velocity and $N_t = 2$ as the number of time steps for computing the characteristic [36].

Table 3

The performance of different solvers for NEW, O1, *and* O2 *in Test* 1 *by using the same stopping criteria. Here, MINRES, MINRESD, and MINRESL represent that the solver is MINRES without preconditioner, with diagonal preconditioner, and with L preconditioner, respectively. CG, CGD, and CGL represent that the solver is CG without preconditioner, with diagonal preconditioner, and with L preconditioner, respectively. And GMRES, GMRESD, and GMRESL represent that the solver is GMRES without preconditioner, with diagonal preconditioner, and with L preconditioner, respectively.*

	NEW			O1			O2		
	Iter on levels	Re_SSD	Time (s)	Iter on levels	Re_SSD	Time (s)	Iter on levels	Re_SSD	Time (s)
MINRES	11,5,3,5	0.10%	21.9	10,5,3,4	0.11%	17.6	12,3,1,8	0.47%	50.6
MINRESD	11,5,4,3	0.22%	13.2	11,4,3,3	0.21%	12.9	12,5,4,4	0.13%	14.6
MINRESL	13,5,3,4	0.08%	15.2	10,4,3,4	0.11%	15.5	11,4,3,4	0.12%	14.4
CG	11,4,3,3	0.22%	15.9	11,4,3,3	0.22%	15.3	13,4,4,5	0.14%	32.3
CGD	11,3,3,5	0.11%	24.6	11,3,3,3	0.23%	13.6	14,4,4,5	0.12%	22.3
CGL	12,4,3,4	0.11%	17.2	12,4,3,3	0.21%	13.4	14,4,5,5	0.12%	21.1
GMRES	11,5,3,5	0.10%	22.4	10,5,3,4	0.11%	17.7	12,3,1,8	0.47%	53.7
GMRESD	12,4,3,4	0.19%	17.3	12,4,3,4	0.11%	18.0	13,5,4,3	0.23%	14.2
GMRESL	12,4,3,4	0.11%	17.6	10,4,3,4	0.11%	17.2	11,5,4,4	0.12%	18.3

Table 4

Performance of the solvers based on CG for O1 *in Test* 1 *by using a strict stopping criterion.*

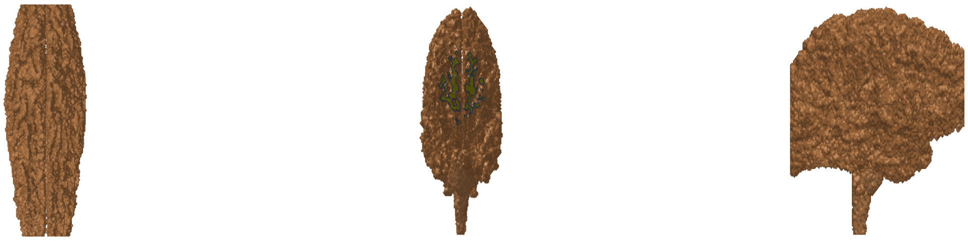
	NEW		
	Iter on levels	Re_SSD	Time (s)
CG	11,5,4,4	0.14%	21.5
CGD	11,4,4,5	0.12%	22.4
CGL	12,4,3,4	0.14%	17.1

For NEW, O1, and O2, we again fix $\alpha_1 = 100$, $\alpha_2 = 0.1$ and choose $\beta = 5000$, $\beta = 5000$, and $\beta = 70$, respectively.

Figure 9 shows the deformed templates obtained by these models, and Table 5 gives the corresponding quantitative measurements. Similar to Test 1 results, we observe that although the deformed templates obtained by these five methods are visually good and the resulting transformations are all orientation-preserving (since the minimums of the Jacobian determinant of the transformations are positive), NEW gives the smallest relative residual. In addition, NEW also produces the best Dice among these models. Specifically, NEW, O1, and O2 need far fewer iterations than Hyper, and their total running times are about half of or less than half of those of Hyper and LDDMM.

5.4. Test 3: Comparison of models for a set of MR images. In this test, we use a set of 17 MR images from the Internet Brain Segmentation Repository (IBSR) to test our new model NEW and the other two models (Hyper and LDDMM). The data are downloaded from https://www.nitrc.org/projects/ibsr. Here, we resize the images into $128 \times 128 \times 128$. We fix one of the images as the template and take the other images as the references. Hence, we in total have 16 pairs of templates and references. The domain of the images is $[0, 1]^3$. In the implementation, for all three models, we employ a five-step multilevel strategy, which is to discretize the images in the following different resolutions: $8 \times 8 \times 8$, $16 \times 16 \times 16$, $32 \times 32 \times 32$, $64 \times 64 \times 64$, and $128 \times 128 \times 128$. The number of unknowns on the finest level in this example is 6440067, making the task a large scale computing problem.

(a) Template T (b) Reference R



(c) Template T in axial, coronal, and sagittal views

(d) Reference R in axial, coronal, and sagittal views

Figure 8. *Test* 2: *The first line shows the template and reference. The second and third lines show the template and the reference in axial, coronal, and sagittal views, respectively.*

For the parameters of these three models, we test six different parameters, respectively. For the parameters of Hyper, we set $\alpha_l = 1000$ or 100, $\alpha_s = 10$ or 100, and $\alpha_v = 10$ or 100. For

Table 5
Test 2: Comparison of the new models with Hyper and LDDMM.

	Resolution	Re_SSD	$\min\det\nabla\boldsymbol{y}$	$\max\det\nabla\boldsymbol{y}$	DSC	Time (s)	Iterations on each level
NEW	$128\times 64\times 128$	8.12%	0.0097	39.3644	0.8632	295.7	$9, 11, 13, 16$
O1	$128\times 64\times 128$	11.93%	0.0447	36.6370	0.8552	188.9	$7, 11, 13, 10$
O2	$128\times 64\times 128$	9.95%	0.0615	43.9114	0.8583	221.7	$5, 9, 11, 14$
Hyper	$128\times 64\times 128$	11.33%	0.0026	4.6357	0.8555	580.4	$8, 9, 16, 24$
LDDMM	$128\times 64\times 128$	18.59%	0.0032	17.8784	0.8422	773.6	$3, 5, 7, 8$

LDDMM (see [36] for a similar choice of parameters), we vary α from 100 to 1000 to control the smoothness of the velocity and set $N_t = 2$ as the number of time steps for computing the characteristic. For NEW, we fix $\alpha_1 = 100$ and set $\alpha_2 = 0.01, 0.1,$ or 1 and $\beta = 5000$ or 10000.

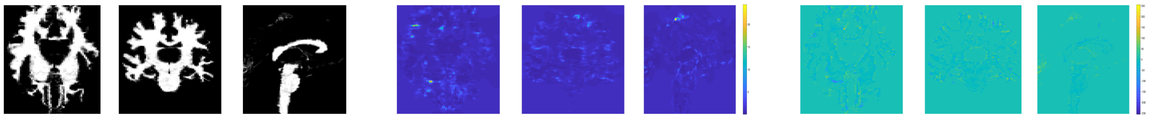
Figure 10 shows the deformed templates, relative volume changes, and residuals of one case obtained by these three models. The corresponding measurements are shown in Table 6. Here, to reflect results from these pairs of images, we list the average and standard deviation of the Re_SSD, $\min\det\nabla\boldsymbol{y}$, $\max\det\nabla\boldsymbol{y}$ and computational time of different methods with respect to different values of the regularization parameters. All the methods can guarantee the bijective transformations because all the minimums of the Jacobian determinant of the transformations are positive. For Hyper, by choosing these parameters, the ranges of the Jacobian determinant of the transformation are very similar. This is because Hyper has a potential to force $\det\nabla\boldsymbol{y} \approx 1$. However, compared with NEW and LDDMM, the relative SSD obtained by Hyper is worse, which shows that preserving volume is not suitable in this application. Further, NEW can give better Re_SSD than LDDMM. Especially when the parameters are set $(100, 0.1, 5000)$, NEW can generate the best Re_SSD among all these choices. Here, we note that when the parameters are set $(100, 0.01, 5000)$, the generated Re_SSD is slighter worse than $(100, 0.1, 5000)$, but the computational time is only about 50%. For LDDMM, by tuning the parameters, we can get acceptable Re_SSD, but it needs much more running time than NEW.

In summary, the above three sets of examples demonstrate that our new model NEW can be more advantageous than (and competitive with) the state-of-the-art models, Hyper, LDDMM, and O2 (LLL) in terms of computational time and accuracy.

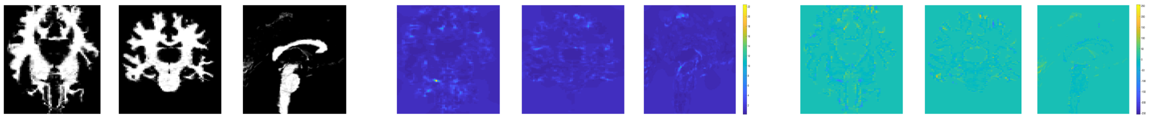
6. Conclusions.

In image registration, visual comparison is not a reliable way to assess effectiveness because our human eye cannot always tell if a transformed image is incorrect due to going through a folding transformation. In order to ensure that the transformation has no folding, many models (including the state-of-the-art registration models) explicitly control the Jacobian determinant of the underlying transformation. However, for registration problems where larger deformations exist, controlling the Jacobian determinant of the transformation and forcing it to be close to 1 are not always reasonable. To overcome this difficulty, minimizing the modulus of the Beltrami coefficient offers an indirect way of controlling the Jacobian determinant of the transformation. However, since the Beltrami coefficient is defined in two dimensions and by complex analysis, it cannot be directly extended to 3D. In this paper, we construct a quantity as a generalization of the norm of the Beltrami coefficient in 3D as a measure of distortion on conformal maps. Using it, we propose our new model (18) and establish the existence of a solution. In order to solve the new model efficiently, we present

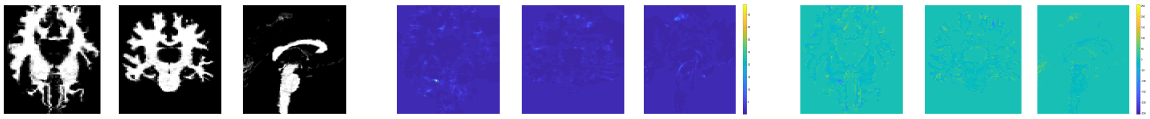
(a) Template (b) Reference (c) Residual before registration



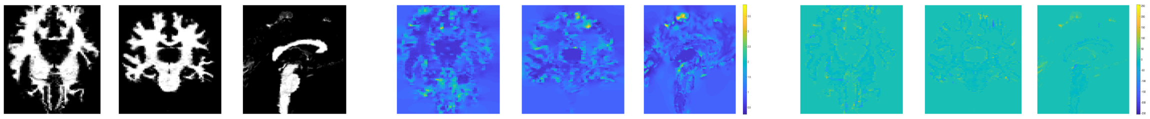
(d) $T(\boldsymbol{y}_{\mathrm{NEW}})$ (e) Relative volume change (f) Residual after registration (8.12%)



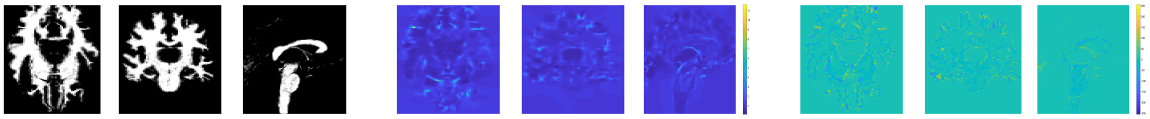
(g) $T(\boldsymbol{y}_{\mathrm{O1}})$ (h) Relative volume change (i) Residual after registration (11.93%)



(j) $T(\boldsymbol{y}_{\mathrm{O2}})$ (k) Relative volume change (l) Residual after registration (9.95%)



(m) $T(\boldsymbol{y}_{\mathrm{Hyper}})$ (n) Relative volume change (o) Residual after registration (11.33%)



(p) $T(\boldsymbol{y}_{\mathrm{LDDMM}})$ (q) Relative volume change (r) Residual after registration (18.59%)

Figure 9. *The results of Test* 2*: In the first row, there are the template, reference, and the residual before registration in axial, coronal, and sagittal views. The second row to the sixth row show the deformed template, its corresponding relative volume change* ($\det \nabla \boldsymbol{y}$)*, and residual after registration in axial, coronal, and sagittal views obtained by NEW,* O1*,* O2*, Hyper, and LDDMM, respectively. The percentage represents the relative residual.*

Table 6

Test 3 of 16 pairs of MR images: Comparison of the new model NEW with Hyper and LDDMM. Average and standard deviation of the Re_SSD, $\min\det\nabla\boldsymbol{y}$, $\max\det\nabla\boldsymbol{y}$, *and computing time for different values of the regularization parameters.*

Measurements by NEW

Parameters	$\min\det\nabla\boldsymbol{y}$	$\max\det\nabla\boldsymbol{y}$	Re_SSD	Time (s)
(100, 0.01, 5000)	0.1398 ± 0.0743	11.4426 ± 9.8745	$12.59\% \pm 4.61\%$	408.7 ± 130.9
(100, 0.01, 10000)	0.1864 ± 0.0682	9.0047 ± 7.2030	$14.33\% \pm 5.51\%$	430.5 ± 177.1
(100, 0.1, 5000)	0.1800 ± 0.1043	6.8892 ± 5.5417	$12.24\% \pm 5.99\%$	821.8 ± 1167.4
(100, 0.1, 1000)	0.2348 ± 0.0862	5.7243 ± 4.0364	$13.18\% \pm 6.68\%$	1323.6 ± 2551.4
(100, 1, 5000)	0.4060 ± 0.1813	1.7849 ± 2.6414	$28.09\% \pm 9.80\%$	417.4 ± 100.1
(100, 1, 10000)	0.4466 ± 0.1437	1.7761 ± 2.1036	$28.87\% \pm 9.92\%$	357.3 ± 101.7

Measurements by Hyper

Parameters	$\min\det\nabla\boldsymbol{y}$	$\max\det\nabla\boldsymbol{y}$	Re_SSD	Time (s)
(100, 10, 100)	0.3574 ± 0.0526	2.5507 ± 0.6171	$17.44\% \pm 8.59\%$	1197.7 ± 1243.5
(100, 100, 10)	0.2721 ± 0.0823	1.6649 ± 0.2513	$25.39\% \pm 14.04\%$	2920.7 ± 1686.1
(100, 100, 100)	0.4236 ± 0.0640	1.6509 ± 0.2680	$25.89\% \pm 14.04\%$	1662.1 ± 634.5
(1000, 10, 100)	0.4149 ± 0.0458	2.2362 ± 0.3953	$21.20\% \pm 9.75\%$	347.9 ± 216.7
(1000, 100, 10)	0.4143 ± 0.0983	1.5845 ± 0.1734	$33.04\% \pm 12.79\%$	262.7 ± 63.8
(1000, 100, 100)	0.4986 ± 0.0678	1.5788 ± 0.1700	$33.18\% \pm 12.77\%$	271.7 ± 67.5

Measurements by LDDMM

Parameters	$\min\det\nabla\boldsymbol{y}$	$\max\det\nabla\boldsymbol{y}$	Re_SSD	Time (s)
100	0.0659 ± 0.0632	45.6814 ± 61.2808	$13.17\% \pm 5.20\%$	2328.4 ± 1564.6
200	0.1162 ± 0.0861	13.6115 ± 13.7689	$15.89\% \pm 5.64\%$	1997.2 ± 1794.0
400	0.2024 ± 0.1099	5.7173 ± 3.8898	$20.49\% \pm 7.35\%$	1920.8 ± 2150.8
600	0.2834 ± 0.1198	4.0926 ± 2.2623	$23.53\% \pm 8.46\%$	1410.4 ± 1693.6
800	0.3438 ± 0.1282	3.3744 ± 1.5352	$25.66\% \pm 9.15\%$	1215.6 ± 1558.0
1000	0.4002 ± 0.1255	3.0326 ± 1.2210	$27.40\% \pm 9.67\%$	999.0 ± 1088.4

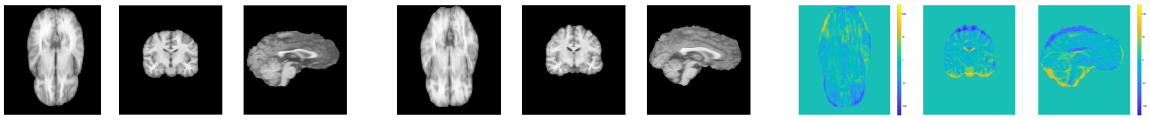
a converging generalized Gauss–Newton scheme. The numerical experiments illustrate that our new model can be more advantageous than related models with respect to computational time and accuracy.

In the future, we will consider a possible reformation by the game approach [46] to reduce the number of model parameters and test the new model for multimodal images. We also hope to develop an unsupervised deep learning method following the approach of [11, 47], where the energy functional (18) in our proposed registration model is used as a loss function (without using any ground truth data). Finally, there is also a recent development in hypercomplex analysis using Clifford analysis (to extend 2D complex analysis to higher dimensions). It would be of interest to consider how to generalize the 2D Beltrami coefficient in this framework.

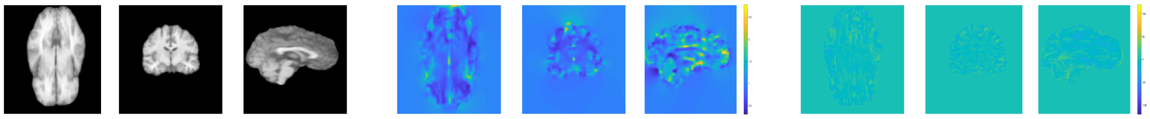
Appendix A. Computation of A in (29).

$$A = I_3 \otimes \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix}, \tag{63}$$

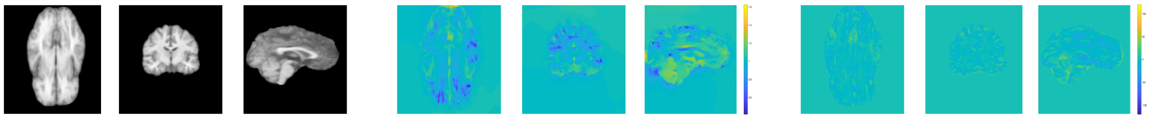
where $A_1 = I_{(n_3+1)} \otimes I_{(n_2+1)} \otimes \partial_{n_1}^{1,h_1}$, $A_2 = I_{(n_3+1)} \otimes \partial_{n_2}^{1,h_2} \otimes I_{(n_1+1)}$, $A_3 = \partial_{n_3}^{1,h_3} \otimes I_{(n_2+1)} \otimes I_{(n_1+1)}$,



(a) Template (b) Reference (c) Residual before registration



(d) $T(\boldsymbol{y}_{\text{NEW}})$ (e) Relative volume change (f) Residual after registration (7.73%)



(g) $T(\boldsymbol{y}_{\text{Hyper}})$ (h) Relative volume change (i) Residual after registration (9.43%)

(j) $T(\boldsymbol{y}_{\text{LDDMM}})$ (k) Relative volume change (l) Residual after registration (9.35%)

Figure 10. *The results of one case in Test* 3*: In the first row, there are the template, reference, and the residual before registration in axial, coronal, and sagittal views. The second row to the fourth row show the deformed template, its corresponding relative volume change* ($\det \nabla \boldsymbol{y}$)*, and residual after registration in axial, coronal, and sagittal views obtained by NEW, Hyper, and LDDMM, respectively. The percentage represents the relative residual. Here, the parameters of NEW, Hyper, and LDDMM are* (100,0.01,5000)*,* (100,100,10)*, and* 200*, respectively.*

and

$$\partial_{n_l}^{1,h_l} = \frac{1}{h_l}\begin{pmatrix} -1 & 1 & & \\ & \cdot & \cdot & \\ & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{n_l, n_l+1}, \quad 1 \le l \le 3. \tag{64}$$

Here, $\otimes$ indicates Kronecker product.

Appendix B. Computation of B in (30).

$$B = I_3 \otimes \begin{pmatrix} B_1^T & B_2^T & B_3^T & B_4^T & B_5^T & B_6^T & B_7^T & B_8^T & B_9^T \end{pmatrix}^T, \tag{65}$$

where $B_1 = I_{(n_3+1)} \otimes I_{(n_2+1)} \otimes \partial_{n_1}^{2,h_1}$, $B_2 = I_{(n_3+1)} \otimes \partial_{n_2}^{2,h_2} \otimes I_{(n_1+1)}$, $B_3 = \partial_{n_3}^{2,h_3} \otimes I_{(n_2+1)} \otimes I_{(n_1+1)}$, $B_4 = B_7 = I_{(n_3+1)} \otimes \partial_{n_2}^{1,h_2} \otimes \partial_{n_1}^{1,h_1}$, $B_5 = B_8 = \partial_{n_3}^{1,h_3} \otimes I_{(n_2+1)} \otimes \partial_{n_1}^{1,h_1}$, $B_6 = B_9 = \partial_{n_3}^{1,h_3} \otimes \partial_{n_2}^{1,h_2} \otimes$

$I_{(n_1+1)}$, and

$$\partial_{n_l}^{2,h_l} = \frac{1}{h_l^2}\begin{pmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \cdot & \cdot & \\ & & 1 & -2 \end{pmatrix} \in \mathbb{R}^{n_l+1,n_l+1}, \quad 1 \le l \le 3. \tag{66}$$

Here, $\otimes$ indicates Kronecker product.

Appendix C. Computation of M_1, M_2, and M_3 in (34). We first investigate the linear approximation $L(x_1, x_2, x_3) = a_1x_1 + a_2x_2 + a_3x_3 + b$ in the tetrahedron $V_3V_4V_5V_7$ (Figure 2). Denote these four vertices of this tetrahedron by $V_3 = \boldsymbol{x}^{1,1,1}$, $V_4 = \boldsymbol{x}^{2,2,2}$, $V_5 = \boldsymbol{x}^{3,3,3}$, and $V_7 = \boldsymbol{x}^{4,4,4}$. Set $L(\boldsymbol{x}^{1,1,1}) = y^{1,1,1}$, $L(\boldsymbol{x}^{2,2,2}) = y^{2,2,2}$, $L(\boldsymbol{x}^{3,3,3}) = y^{3,3,3}$, and $L(\boldsymbol{x}^{4,4,4}) = y^{4,4,4}$. Substituting V_3, V_4, V_5, and V_7 into L, we get

$$\begin{pmatrix} x_1^1 & x_2^1 & x_3^1 & 1 \\ x_1^2 & x_2^2 & x_3^2 & 1 \\ x_1^3 & x_2^3 & x_3^3 & 1 \\ x_1^4 & x_2^4 & x_3^4 & 1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ b \end{pmatrix} = \begin{pmatrix} y^{1,1,1} \\ y^{2,2,2} \\ y^{3,3,3} \\ y^{4,4,4} \end{pmatrix}. \tag{67}$$

Then, eliminating b, we obtain

$$\begin{pmatrix} x_1^1 - x_1^4 & x_2^1 - x_2^4 & x_3^1 - x_1^4 \\ x_1^2 - x_2^4 & x_2^2 - x_2^4 & x_3^2 - x_2^4 \\ x_1^3 - x_3^4 & x_2^3 - x_2^4 & x_3^3 - x_3^4 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y^{1,1,1} - y^{4,4,4} \\ y^{2,2,2} - y^{4,4,4} \\ y^{3,3,3} - y^{4,4,4} \end{pmatrix}. \tag{68}$$

Set

$$C = \begin{pmatrix} x_1^1 - x_1^4 & x_2^1 - x_2^4 & x_3^1 - x_1^4 \\ x_1^2 - x_2^4 & x_2^2 - x_2^4 & x_3^2 - x_2^4 \\ x_1^3 - x_3^4 & x_2^3 - x_2^4 & x_3^3 - x_3^4 \end{pmatrix}. \tag{69}$$

Then we have

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \frac{1}{\det}\begin{pmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{pmatrix} \begin{pmatrix} y^{1,1,1} - y^{4,4,4} \\ y^{2,2,2} - y^{4,4,4} \\ y^{3,3,3} - y^{4,4,4} \end{pmatrix}, \tag{70}$$

where det is the determinant of C and C_{ij} is the (i, j) cofactor of C. Since the domain Ω has been divided into N voxels, in order to find all a_1 in the tetrahedron with the same position of each voxel, we can make it as follows:

$$\begin{pmatrix} a_1^1 \\ \vdots \\ a_1^N \end{pmatrix} = \frac{1}{\det}(C_{11}(E_3Y - E_7Y) + C_{21}(E_4Y - E_7Y) + C_{31}(E_5Y - E_7Y)), \tag{71}$$

where $E_l, l \in \{3, 4, 5, 7\}$, is a matrix which extracts the corresponding positions of the vertices. Set $G_1 = \frac{1}{\det}(C_{11}(E_3 - E_7) + C_{21}(E_4 - E_7) + C_{31}(E_5 - E_7))$. For the other five tetrahedrons, we can also build $G_l, l \in \{2, \dots, 6\}$. Then we get

$$M_1 = \begin{pmatrix} G_1 \\ \vdots \\ G_6 \end{pmatrix}. \tag{72}$$

Similarly, we can obtain M_2 and M_3.

Appendix D. Computation of the matrix-vector product $\hat{H}v$. Recall that $\hat{H} = \hat{H}_1 + H_2 + \hat{H}_3$ and we have $\hat{H}v = \hat{H}_1 v + H_2 v + \hat{H}_3 v$.

First, for $\hat{H}_1 v = hP^T \vec{T}^T_{\tilde{\mathbf{Y}}} \vec{T}_{\tilde{\mathbf{Y}}} Pv$, we need to compute $v_1 = Pv$, $v_2 = \vec{T}_{\tilde{\mathbf{Y}}} v_1$, $v_3 = \vec{T}^T_{\tilde{\mathbf{Y}}} v_2$, and $\hat{H}_1 v = P^T v_3$. Since P is an averaging matrix from the nodal grid to the cell-centered grid, as an example, the first component of Pv is

$$\begin{aligned}(Pv)_1 =& \frac{1}{8}((v)_1 + (v)_2 + (v)_{1+n_1} + (v)_{2+n_1} + (v)_{1+(n_1+1)(n_2+1)} \\ &+ (v)_{2+(n_1+1)(n_2+1)} + (v)_{1+n_1+(n_1+1)(n_2+1)} + (v)_{2+n_1+(n_1+1)(n_2+1)}).\end{aligned} \tag{73}$$

$\vec{T}_{\tilde{\mathbf{Y}}}$ has the following structure:

$$\vec{T}_{\tilde{\mathbf{Y}}} = [\mathrm{diag}(w_1), \mathrm{diag}(w_2), \mathrm{diag}(w_3)]. \tag{74}$$

Then we have $\vec{T}_{\tilde{\mathbf{Y}}} v_1 = \Sigma_{l=1}^3 w_l \odot v_{1l}$ and $\vec{T}^T_{\tilde{\mathbf{Y}}} v_2 = ((w_1 \odot v_2)^T, (w_2 \odot v_2)^T, (w_3 \odot v_2)^T)^T$, where $v_1 = (v_{11}^T, v_{12}^T, v_{13}^T)^T$. Similarly, it is easy to implement $P^T v_3$.

Second, in order to compute $H_2 v = (\alpha_1 h A^T A + \alpha_2 h B^T B)v$, we just consider how A_l and $A_l^T, l \in \{1, 2, 3\}$, multiply a vector and B_l and $B_l^T, l \in \{1, \dots, 9\}$, multiply a vector. According to (63) and (65), because of $\partial_{n_l}^{2,h_l} = (\partial_{n_l}^{2,h_l})^T$, here we only need to investigate $\partial_{n_l}^{1,h_l} v', (\partial_{n_l}^{1,h_l})^T v'$ and $\partial_{n_l}^{2,h_l} v'$, $l = \{1, 2, 3\}$.

Finally, because $\hat{H}_3 = \frac{\beta h}{6} \mathrm{d}\vec{\boldsymbol{r}}^T \mathrm{d}^2 \boldsymbol{\phi}(\vec{\boldsymbol{r}}) \mathrm{d}\vec{\boldsymbol{r}}$ and $\mathrm{d}^2 \boldsymbol{\phi}(\vec{\boldsymbol{r}})$ is a diagonal matrix, we only need to consider computing $\mathrm{d}\vec{\boldsymbol{r}} v$ and $\mathrm{d}\vec{\boldsymbol{r}}^T v'$. According to the (35), substituting $\mathrm{d}\vec{\boldsymbol{r}}^1$, $\mathrm{d}\vec{\boldsymbol{r}}^2$, $\mathrm{d}\vec{\boldsymbol{q}}^1$, and $\mathrm{d}\vec{\boldsymbol{q}}^2$ into $\mathrm{d}\vec{\boldsymbol{r}}$, we have

$$\mathrm{d}\vec{\boldsymbol{r}} = \sum_{l=1}^{9} \Lambda_l D_l, \tag{75}$$

where

$$\begin{aligned}
\Lambda_1 &= 2\Gamma_1 \,\mathrm{diag}(D_1 Y) + 2\Gamma_2 \,\mathrm{diag}(D_5 Y \odot D_9 Y - D_6 Y \odot D_8 Y), \\
\Lambda_2 &= 2\Gamma_1 \,\mathrm{diag}(D_2 Y) + 2\Gamma_2 \,\mathrm{diag}(D_6 Y \odot D_7 Y - D_4 Y \odot D_9 Y), \\
\Lambda_3 &= 2\Gamma_1 \,\mathrm{diag}(D_3 Y) + 2\Gamma_2 \,\mathrm{diag}(D_4 Y \odot D_8 Y - D_5 Y \odot D_7 Y), \\
\Lambda_4 &= 2\Gamma_1 \,\mathrm{diag}(D_4 Y) + 2\Gamma_2 \,\mathrm{diag}(D_8 Y \odot D_3 Y - D_2 Y \odot D_9 Y), \\
\Lambda_5 &= 2\Gamma_1 \,\mathrm{diag}(D_5 Y) + 2\Gamma_2 \,\mathrm{diag}(D_1 Y \odot D_9 Y - D_3 Y \odot D_7 Y), \\
\Lambda_6 &= 2\Gamma_1 \,\mathrm{diag}(D_6 Y) + 2\Gamma_2 \,\mathrm{diag}(D_2 Y \odot D_7 Y - D_1 Y \odot D_8 Y), \\
\Lambda_7 &= 2\Gamma_1 \,\mathrm{diag}(D_7 Y) + 2\Gamma_2 \,\mathrm{diag}(D_2 Y \odot D_6 Y - D_3 Y \odot D_5 Y), \\
\Lambda_8 &= 2\Gamma_1 \,\mathrm{diag}(D_8 Y) + 2\Gamma_2 \,\mathrm{diag}(D_4 Y \odot D_3 Y - D_1 Y \odot D_6 Y), \\
\Lambda_9 &= 2\Gamma_1 \,\mathrm{diag}(D_9 Y) + 2\Gamma_2 \,\mathrm{diag}(D_1 Y \odot D_5 Y - D_2 Y \odot D_4 Y),
\end{aligned} \tag{76}$$

$\Gamma_1 = -\frac{1}{2}\,\mathrm{diag}((\vec{\boldsymbol{r}}^1 \odot \vec{\boldsymbol{r}}^2 \odot \vec{\boldsymbol{r}}^2 - \vec{\boldsymbol{r}}^2)./(\vec{\boldsymbol{q}}^1)^{\frac{1}{2}})$, and $\Gamma_2 = -\frac{\sqrt{3}}{3}\,\mathrm{diag}((\vec{\boldsymbol{r}}^1 \odot \vec{\boldsymbol{r}}^2 \odot \vec{\boldsymbol{r}}^2 + \vec{\boldsymbol{r}}^2)./(\vec{\boldsymbol{q}}^2)^{\frac{2}{3}})$. Furthermore, because of (34), (75) can be reformulated as follows:

$$\mathrm{d}\vec{\boldsymbol{r}} = [\Lambda_1 M_1 + \Lambda_2 M_2 + \Lambda_3 M_3, \Lambda_4 M_1 + \Lambda_5 M_2 + \Lambda_6 M_3, \Lambda_7 M_1 + \Lambda_8 M_2 + \Lambda_9 M_3]. \tag{77}$$

Hence, we only need to compute $M_l v_k$, where $l, k \in \{1,2,3\}$ and $v = (v_1^T, v_2^T, v_3^T)^T$. For simplification, we only consider $M_1 v_1$. Recalling (72), we can get

$$M_1 v_1 = \begin{pmatrix} G_1 v_1 \\ \vdots \\ G_6 v_1 \end{pmatrix}. \tag{78}$$

Since $G_l, l \in \{1,\dots,6\}$, is just the linear combination of the matrix $E_l, l \in \{1,\dots,8\}$, finally we only compute $E_l v_1, l \in \{1,\dots,8\}$, which is very easy to implement.

Similarly, in order to compute $\mathrm{d}\vec{\boldsymbol{r}}^T v'$, we only need to compute $M_l^T v', l \in \{1,2,3\}$, and it can be decomposed to compute $E_l^T v_k', l \in \{1,\dots,8\}$ and $k \in \{1,\dots,6\}$, where $v' = ((v_1')^T, \dots, (v_6')^T)^T$.

Appendix E. The diagonal of $\hat{H}$ and the preconditioner L. According to the structure of $\hat{H}_1$, the diagonal of $\hat{H}_1$ is $h(P^T \odot P^T)\varsigma$, where ς is the diagonal of $\vec{T}_{\tilde{\mathbf{Y}}}^T \vec{T}_{\tilde{\mathbf{Y}}}$.

The diagonal of H_2 is $\alpha_1 h(A^T \odot A^T)e + \alpha_2 h(B^T \odot B^T)e$, where e is a vector whose components are all equal to 1.

From (77) and $\hat{H}_3 = \frac{\beta h}{6}\mathrm{d}\vec{\boldsymbol{r}}^T \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\mathrm{d}\vec{\boldsymbol{r}}$, the diagonal of $\hat{H}_3$ is $\frac{\beta h}{6}(\varsigma_1^T, \varsigma_2^T, \varsigma_3^T)^T$, where

$$\begin{aligned} \varsigma_1 &= \text{the diagonal of } (\Lambda_1 M_1 + \Lambda_2 M_2 + \Lambda_3 M_3)^T \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})(\Lambda_1 M_1 + \Lambda_2 M_2 + \Lambda_3 M_3), \\ \varsigma_2 &= \text{the diagonal of } (\Lambda_4 M_1 + \Lambda_5 M_2 + \Lambda_6 M_3)^T \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})(\Lambda_4 M_1 + \Lambda_5 M_2 + \Lambda_6 M_3), \\ \varsigma_3 &= \text{the diagonal of } (\Lambda_7 M_1 + \Lambda_8 M_2 + \Lambda_9 M_3)^T \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})(\Lambda_7 M_1 + \Lambda_8 M_2 + \Lambda_9 M_3). \end{aligned} \tag{79}$$

Now we only need to compute the diagonal of $M_{i_1}^T \Lambda_{j_1} \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\Lambda_{j_2} M_{i_2}$, where $i_1, i_2 \in \{1,2,3\}$ and $j_1, j_2 \in \{1,2,3\}, \{4,5,6\}$, or $\{7,8,9\}$. Since $\Lambda_{j_1}\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\Lambda_{j_2}$ is a diagonal matrix and set ς is the diagonal of $\Lambda_{j_1}\mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\Lambda_{j_2}$, the diagonal of $M_{i_1}^T \Lambda_{j_1} \mathrm{d}^2\boldsymbol{\phi}(\vec{\boldsymbol{r}})\Lambda_{j_2} M_{i_2}$ is $(M_{i_1}^T \odot M_{i_2}^T)\varsigma$, which is very easy to implement following Appendix D.

The structure of the preconditioner L is

$$\begin{pmatrix} \mathrm{diag}(\hat{H}_{11}) & \mathrm{diag}(\hat{H}_{12}) & \mathrm{diag}(\hat{H}_{13}) \\ \mathrm{diag}(\hat{H}_{21}) & \mathrm{diag}(\hat{H}_{22}) & \mathrm{diag}(\hat{H}_{23}) \\ \mathrm{diag}(\hat{H}_{31}) & \mathrm{diag}(\hat{H}_{32}) & \mathrm{diag}(\hat{H}_{33}) \end{pmatrix}. \tag{80}$$

Since $\hat{H}$ is symmetric and we have obtained the diagonal of $\hat{H}$, we only need to compute $\mathrm{diag}(\hat{H}_{12})$, $\mathrm{diag}(\hat{H}_{13})$, and $\mathrm{diag}(\hat{H}_{23})$. Actually, they are also computed easily just by following the above-mentioned steps.

REFERENCES

[1] N. Alexander, S. Emil, and Z. Y. Yehoshua, *Computing quasi-conformal maps in 3D with applications to geometric modeling and imaging*, in Proceedings of the 28th IEEE Convention of Electrical & Electronics Engineers in Israel (IEEEI), IEEE, Washington, DC, 2014, pp. 1–5.

[2] J. M. Ball, *Global invertibility of Sobolev functions and the interpenetration of matter*, Proc. Roy. Soc. Edinburgh Sect. A, 88 (1981), pp. 315–328.

[3] R. Barrett, M. W. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994, https://doi.org/10.1137/1.9781611971538.

[4] M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes, *Computing large deformation metric mappings via geodesic flows of diffeomorphisms*, Internat. J. Comput. Vis., 61 (2005), pp. 139–157.

[5] L. Bers, *Quasiconformal mappings, with applications to differential equations, function theory and topology*, Bull. Amer. Math. Soc., 83 (1977), pp. 1083–1100.

[6] C. Broit, *Optimal Registration of Deformed Images*, Ph.D. thesis, University of Pennsylvania, Philadelphia, PA, 1981.

[7] L. G. Brown, *A survey of image registration techniques*, ACM Comput. Surveys, 24 (1992), pp. 325–376.

[8] M. Burger, J. Modersitzki, and L. Ruthotto, *A hyperelastic regularization energy for image registration*, SIAM J. Sci. Comput., 35 (2013), pp. B132–B148, https://doi.org/10.1137/110835955.

[9] C. Chen and O. Öktem, *Indirect image registration with large diffeomorphic deformations*, SIAM J. Imaging Sci., 11 (2018), pp. 575–617, https://doi.org/10.1137/17M1134627.

[10] K. Chen, G. N. Grapiglia, J. Yuan, and D. Zhang, *Improved optimization methods for image registration problems*, Numer. Algorithms, 80 (2019), pp. 305–336.

[11] K. Chen, L. M. Lui, and J. Modersitzki, *Image and surface registration*, in Processing, Analyzing and Learning of Images, Shapes, and Forms, Part 2, Handb. Numer. Anal. 20, Elsevier/North-Holland, Amsterdam, 2019, pp. 579–611.

[12] G. E. Christensen, R. D. Rabbitt, and M. I. Miller, *Deformable templates using large deformation kinematics*, IEEE Trans. Image Process., 5 (1996), pp. 1435–1447.

[13] N. Chumchob and K. Chen, *A variational approach for discontinuity-preserving image registration*, East-West J. Math., Special Vol. (2010), pp. 266–282.

[14] N. Chumchob, K. Chen, and C. Brito-Loeza, *A fourth-order variational image registration model and its fast multigrid algorithm*, Multiscale Model. Simul., 9 (2011), pp. 89–128, https://doi.org/10.1137/100788239.

[15] M. Droske and M. Rumpf, *A variational approach to nonrigid morphological image registration*, SIAM J. Appl. Math., 64 (2004), pp. 668–687, https://doi.org/10.1137/S0036139902419528.

[16] P. Dupuis, U. Grenander, and M. I. Miller, *Variational problems on flows of diffeomorphisms for image matching*, Quart. Appl. Math., 56 (1998), pp. 587–600.

[17] L. C. Evans, *Partial Differential Equations*, 2nd ed., Grad. Stud. Math. 19, AMS, Providence, RI, 2010.

[18] G. J. M. F. W. Gehring, and B. P. Palka, *An Introduction to the Theory of Higher-Dimensional Quasiconformal Mappings*, AMS, Providence, RI, 2017.

[19] B. Fischer and J. Modersitzki, *Fast diffusion registration*, in Inverse Problems, Image Analysis, and Medical Imaging, Contemp. Math. 313, AMS, Providence, RI, 2002, pp. 117–127.

[20] B. Fischer and J. Modersitzki, *Curvature based image registration*, J. Math. Imaging Vis., 18 (2003), pp. 81–85.

[21] B. Fischer and J. Modersitzki, *A unified approach to fast image registration and a new curvature based registration technique*, Linear Algebra Appl., 380 (2004), pp. 107–124.

[22] C. Frohn-Schauf, S. Henn, and K. Witsch, *Multigrid based total variation image registration*, Comput. Vis. Sci., 11 (2008), pp. 101–113.

[23] E. Haber and J. Modersitzki, *Numerical methods for volume preserving image registration*, Inverse Problems, 20 (2004), pp. 1621–1638.

[24] E. Haber and J. Modersitzki, *Intensity gradient based registration and fusion of multi-modal images*, in Medical Image Computing and Computer-Assisted Intervention—MICCAI 2006, Springer, New York, 2006, pp. 726–733.

[25] E. Haber and J. Modersitzki, *Image registration with guaranteed displacement regularity*, Internat. J. Comput. Vis., 71 (2007), pp. 361–372.

[26] E. Haber and J. Modersitzki, *Intensity gradient based registration and fusion of multi-modal images*, Methods Inform. Med., 46 (2007), pp. 292–299.

[27] J. V. Hajnal and D. L. Hill, *Medical Image Registration*, CRC Press, Boca Raton, FL, 2001.

[28] M. Ibrahim, K. Chen, and C. Brito-Loeza, *A novel variational model for image registration using Gaussian curvature*, Geom. Imaging Comput., 1 (2014), pp. 417–446.

[29] T. Iwaniec and G. Martin, *Geometric Function Theory and Non-linear Analysis*, Oxford Math. Monogr., Clarendon, Oxford University Press, Oxford, UK, 2001.

[30] C. T. Kelley, *Iterative Methods for Optimization*, Frontiers Appl. Math. 18, SIAM, Philadelphia, 1999, https://doi.org/10.1137/1.9781611970920.

[31] K. C. Lam and L. M. Lui, *Landmark- and intensity-based registration with large deformations via quasi-conformal maps*, SIAM J. Imaging Sci., 7 (2014), pp. 2364–2392, https://doi.org/10.1137/130943406.

[32] Y. T. Lee, K. C. Lam, and L. M. Lui, *Landmark-matching transformation with large deformation via n-dimensional quasi-conformal maps*, J. Sci. Comput., 67 (2016), pp. 926–954.

[33] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, in IJCAI'81: Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vol. 2, ACM, New York, 1981, pp. 674–679.

[34] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, *Multimodality image registration by maximization of mutual information*, IEEE Trans. Med. Imaging, 16 (1997), pp. 187–198.

[35] J. A. Maintz and M. A. Viergever, *A survey of medical image registration*, Med. Image Anal., 2 (1998), pp. 1–36.

[36] A. Mang and L. Ruthotto, *A Lagrangian Gauss–Newton–Krylov solver for mass- and intensity-preserving diffeomorphic image registration*, SIAM J. Sci. Comput., 39 (2017), pp. B860–B885, https://doi.org/10.1137/17M1114132.

[37] G. J. Martin, *The Theory of Quasiconformal Mappings in Higher Dimensions,* I, preprint, https://arxiv.org/abs/1311.0899, 2013.

[38] T. McInerney and D. Terzopoulos, *Deformable models in medical image analysis: A survey*, Med. Image Anal., 1 (1996), pp. 91–108.

[39] J. Modersitzki, *Numerical Methods For Image Registration*, Oxford University Press, Oxford, UK, 2003.

[40] J. Modersitzki, *FAIR: Flexible Algorithms for Image Registration*, Fundam. Algorithms 6, SIAM, Philadelphia, 2009, https://doi.org/10.1137/1.9780898718843.

[41] J. Nocedal and S. Wright, *Numerical Optimization*, Springer, New York, 2006.

[42] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629, https://doi.org/10.1137/0712047.

[43] L. Ruthotto, *Hyperelastic Image Registration: Theory, Numerical Methods, and Applications*, Ph.D. thesis, University of Münster, Münster, Germany, 2012.

[44] A. Sotiras, C. Davatzikos, and N. Paragios, *Deformable medical image registration: A survey*, IEEE Trans. Med. Imaging, 32 (2013), pp. 1153–1190.

[45] A. Theljani and K. Chen, *An augmented Lagrangian method for solving a new variational model based on gradients similarity measures and high order regularization for multimodality registration*, Inverse Probl. Imaging, 13 (2019), pp. 309–335.

[46] A. Theljani and K. Chen, *A Nash game based variational model for joint image intensity correction and registration to deal with varying illumination*, Inverse Problems, 36 (2020), 034002.

[47] A. Theljani and K. Chen, *An unsupervised deep learning method for diffeomorphic mono- and multi-modal image registration*, in Medical Image Understanding and Analysis (MIUA 2019), Communications in Computer and Information Science 1065, Springer, Cham, 2020, pp. 317–326.

[48] A. Trouvé, *Diffeomorphisms groups and pattern matching in image analysis*, Internat. J. Comput. Vis., 28 (1998), pp. 213–221.

[49] Y. Wang, W. Yin, and J. Zeng, *Global convergence of ADMM in nonconvex nonsmooth optimization*, J. Sci. Comput., 78 (2019), pp. 29–63.

[50] D. Zhang and K. Chen, *A novel diffeomorphic model for image registration and its algorithm*, J. Math. Imaging Vis., 60 (2018), pp. 1261–1283.

[51] J. Zhang and K. Chen, *Variational image registration by a total fractional-order variation model*, J. Comput. Phys., 293 (2015), pp. 442–461.

[52] X. Zhou, *Weak lower semicontinuity of a functional with any order*, J. Math. Anal. Appl., 221 (1998), pp. 217–237.

[53] B. Zitova and J. Flusser, *Image registration methods: A survey*, Image Vis. Comput., 21 (2003), pp. 977–1000.