Multi-modality Image Registration Models and Efficient Algorithms

Daoping Zhang, Anis Theljani, and Ke Chen

Dedicated to Professor R. H. Chan on the Occasion of His 60th Birthday.

Abstract In this Chapter we discuss multi-modality image registration models and efficient algorithms. We propose a simple method to enhance a variational model to generate a diffeomorphic transformation. The idea is illustrated by using a particular model based on reformulated normalized gradients of the images as the fidelity term and higher-order derivatives as the regularizer. By adding a control term motivated by quasi-conformal maps and Beltrami coefficients, the model has the ability to guarantee a diffeomorphic transformation. Without this feature, the model may lead to visually pleasing but invalid results. To solve the model numerically, we present both a Gauss-Newton method and an augmented Lagrangian method to solve the resulting discrete optimization problem. A multilevel technique is employed to speed up the initialization and reduce the possibility of getting local minima of the underlying functional. Finally numerical experiments demonstrate that this new model can deliver good performances for multi-modal image registration and simultaneously generate an accurate diffeomorphic transformation.

Keywords Multi-modal image registration · Variational model · Diffeomorphic transformation

AMS subject classifications 65K10 · 68U10

D. Zhang · A. Theljani · K. Chen (✉)
EPSRC Liverpool Centre for Mathematics in Healthcare, Centre for Mathematical Imaging Techniques and Department of Mathematical Sciences, The University of Liverpool, Peach Street, Liverpool L69 7ZL, UK
e-mail: k.chen@liv.ac.uk
URL: https://www.liv.ac.uk/cmit

1 Introduction

Working on a pair of images of the same object taken at different times or acquired using different devices, image registration aims to either find differences between them or fuse complementary information to each other which is otherwise not possible with a single modality. In either case, the key is to find a reasonable spatial geometric transformation between these two images. Though the task is required in diverse fields such as astronomy, optics, biology, chemistry and remote sensing and particularly in medical imaging, and much work have been done, getting a robust model for the task is still a challenge. For an overview of image registration methodologies and approaches, especially for registering images acquired by the same modality (e.g. CT-CT), we refer to [17, 18, 33, 35, 40]. For a more recent survey, see [8]. This Chapter is mainly concerned with registering two images from different modalities (e.g. CT-MRI or digital-Infrared) and focuses on one important question of how to impose a constraint so that the underlying transformation is diffeomorphic.

The image registration problem can be described as follows: given a fixed image R (the reference) and a moving image T (the template), both represented by scalar function mappings over $\Omega \subset \mathbb{R}^d \longrightarrow \mathbb{R}$, find a suitable geometric transformation $\varphi(x) = x + u(x)$, $u : \mathbb{R}^d \longrightarrow \mathbb{R}^d$ such that

$$G_1(T[\varphi]) = G_1(T(x + u(x)) \approx G_2(R), \tag{1}$$

where G_1, G_2 must be chosen suitably in multi-modality scenario, because only features or patterns in T, R visually resemble each other, not their given intensities. In contrast, in mono-modality registration where intensities as well as features in T, R resemble each other, we have $G_i(\cdot) = I_d$, $(i = 1, 2)$ or $T \approx R$ pixel wise. In the special case of parametric models, the solution u (or φ) is assumed to belong to some linear spanned space with known Ansatz functions, depending on few parameters (e.g. affine with 6 parameters in 2D or 12 parameters in 3D). However, not all problems can be solved by parametric models.

Here, we focus on variational models for deformable non-parametric image registration where the unknown u sought in a properly chosen functional space is not assumed to have any parametric forms. The reconstruction problem based on model (1) is an ill-posed inverse problem and thus regularization techniques are needed to overcome ill-posedness [7, 11, 13, 14, 21, 30, 31, 47]. Generally speaking, a regularization technique turns the ill-posed problem (1) into a well-posed optimization model

$$\min_{\mathbf{u} \in \mathcal{H}} \left\{ \mathcal{J}(\mathbf{u}) = S(u) + \frac{\lambda}{2} D(T(x + u), R) \right\} \tag{2}$$

where the displacement u is a minimizer of the above joint energy functional and λ is a positive weight which controls the trade-off between them.

In (2), the **first term** $S(\boldsymbol{u})$ is a regularization term which controls the smoothness of $\boldsymbol{u}$ and reflects our expectations in penalizing unlikely transformations. Various regularizers have been proposed, such as first-order derivatives-based on total variation [10, 23], diffusion [15] and elastic regularizer registration models, higher-order derivatives-based on linear curvature [16], mean curvature [12], Gaussian curvature [24], and fractional order derivatives based models [50]; refer also to [11, 31, 44, 51, 52].

The **second term** $D(T(\boldsymbol{x} + \boldsymbol{u}), R)$ is a fidelity measure, which quantifies distance or similarity between the transformed template image $T(\boldsymbol{x} + \boldsymbol{u})$ and the reference R. For mono-modal registration, a widely-used data fidelity term $D(T(\boldsymbol{x} + \boldsymbol{u}), R)$ is the sum of squared differences $D = \|T(\boldsymbol{x} + \boldsymbol{u}) - R\|_2^2 \equiv \mathrm{SSD}(T(\boldsymbol{x} + \boldsymbol{u}), R)$ to measure the difference between the reference image R and the deformed template image $T(\boldsymbol{x} + \boldsymbol{u})$. However for multi-modality registration, the choice of $D(T(\boldsymbol{x} + \boldsymbol{u}), R)$ is more challenging. The main issue is how to design the right (or rather better) similarity measures that can support the difference (in features, colours, gradients, illumination etc.) between images from different modalities (e.g. SSD no longer makes sense). Various measures have been proposed and tested in the literature. Designing a measure which is based on the geometric information such as the gradients of the images is a good choice. See for instance the normalized gradient field (**NGF**) [22, 26, 39], edges sketching registration [1], normalized gradient fitting (**GT**) [22, 43] and Mutual Information [29, 37, 46]. Recently [9] proposed a cross-correlation similarity measure based on reproducing kernel Hilbert spaces and found advantages over Mutual Information.

Many models in the literature, of type (2), do not usually contain constraints to ensure that $\varphi(\boldsymbol{x})$ is a diffeomorphic map for the mono-modal registration. And even fewer theoretical or experimental studies deal with diffeomorphic maps for the multi-modal registration. But non-diffeomorphic maps cause phenomena such as folding or tearing which are usually seen as non-natural transformations between the two images, unless λ is small (implying a poor registration fidelity error). Over the last decade, more and more researchers have focused on diffeomorphic image registration where folding measured by the local invertibility quantity $\det(J_\varphi)$ is reduced or avoided where $\det(J_\varphi)$ is the Jacobian determinant of φ. Under desired assumptions, obtaining a one-to-one mapping is a natural choice, see [7, 14, 19, 20].

After surveying a few models of type (2) for multi-modal images, this Chapter shows how to incorporate a suitable constraint into a model so that it can deliver a diffeomorphic map. We illustrate our idea by a specific model: minimizing a new functional based on using reformulated normalized gradients of the images as the fidelity term [43], higher-order derivatives and a new Beltrami coefficient based term [28, 48]. An effective, iterative scheme is also presented and numerical experimental results show that the new registration model has a good performance.

2 Review of Related Models

For a variational image registration model (2), while there exist many choices for a regularizer $S(\boldsymbol{u})$ such as the diffusion operator or the Laplacian [8], below, we briefly review a few of such choices of $D(T(\boldsymbol{x}+\boldsymbol{u}),R)$ for registering a pair of multi-modal images T, R.

Normalized Gradient Field (NGF) and its variants. The basic idea of **NGF** [22, 26, 39] is the use of a derived information from the image intensity, i.e., the gradient. Similarity measures depending on the gradients or geometry of the images, which naturally encode information about the shape, can be better. The aim is to align the gradients $\nabla T(\boldsymbol{x}+\boldsymbol{u})$ and ∇R by minimizing the cosines distance between them. More precisely, on each point $\boldsymbol{x} \in \Omega$, try to find a displacement $\boldsymbol{u}(x)$ such that $\cos \Theta = 1$ where Θ is the angle between $\nabla T(\boldsymbol{x}+\boldsymbol{u})$ and ∇R, which leads to minimizing the similarity term:

$$D^{NGF}(T(\boldsymbol{x}+\boldsymbol{u}),R) = \int_{\Omega} (1-(\cos \Theta)^2)\, \mathrm{d}\boldsymbol{x} = \int_{\Omega} (1-(\nabla_n T(\boldsymbol{x}+\boldsymbol{u}) \cdot \nabla_n R)^2)\, \mathrm{d}\boldsymbol{x},$$
(3)

where $\nabla_n T(\boldsymbol{x}+\boldsymbol{u}) = \nabla T(\boldsymbol{x}+\boldsymbol{u})/|\nabla T(\boldsymbol{x}+\boldsymbol{u})|$ and $\nabla_n R = \nabla R/|\nabla R|$ are normalized unit vectors. An alternative form of the **NGF** that avoids using terms $\nabla_n T(\boldsymbol{x}+\boldsymbol{u})$ and $\nabla_n R$ which are degenerated in homogeneous regions, reformulate **NGF** as

$$D^{NGF}(T(\boldsymbol{x}+\boldsymbol{u}),R) = \int_{\Omega} (|\nabla T(\boldsymbol{x}+\boldsymbol{u})|^2|\nabla R|^2 - (\nabla T(\boldsymbol{x}+\boldsymbol{u}) \cdot \nabla R)^2)\, \mathrm{d}\boldsymbol{x},\quad (4)$$

Mutual Information (MI). It was firstly proposed in [46] and has been studied in various literatures (see [29, 37]), showcasing its great capability as well as limitations. The basic idea is to compare the histograms of the images by exploiting the following quantity

$$D^{MI}(T(\boldsymbol{x}+\boldsymbol{u}),R) = -\int_{\mathbb{R}^2} p_{T,R}(t,r) \log \frac{p_{T,R}(t,r)}{p_T(t)p_R(r)}\, \mathrm{d}t\mathrm{d}r,$$
(5)

where p_R, p_T are probability distributions of the gray values in R and T, while $p_{T,R}$ is the joint probability of the gray values which can be derived from the joint histogram. The main drawback of **MI** is its sensibility to image quantization and the difficulty in estimating the joint probability density function (PDF). In addition, the measure also fails when two features with different intensities in one image have similar intensities in the other one [27].

Maximum Correlation Coefficient (MCC). It is an extension of well-known Normalized cross correlation (**CC**) measure , which is only efficient for mono-modal images [6, 33], to a measure that is able to handle multi-modal images [9]. The similarity measure is defined by

$$D^{MCC}(T(\boldsymbol{x}+\boldsymbol{u}), R) = (1 - \mathbf{MCC}(T, R))^p := (1 - \max_{f,g} \mathbf{CC}(M, N))^p, \ 0 < p < 1,$$

where $M(\boldsymbol{x}) = f(T(\boldsymbol{x}+\boldsymbol{u}), N(\boldsymbol{x}) = g(R(\boldsymbol{x}))$, f and g are two measurable functions. This **MCC** formulation does not require estimation of the continuous joint PDF and offers a powerful alternative to the models based on maximizing **MI**. However, the computation of the maximum over all functions f and g is a big challenge. The recommended approach in [9] is to approximate it based on the theory of reproducing kernel Hilbert space (**RKHS**) [2, 5].

3 The New Model

We aim to design a variational model building on the energy of the form (2)

$$\min_{\mathbf{u} \in \mathcal{H}} \left\{ \mathcal{J}(\mathbf{u}) = S(\boldsymbol{u}) + D(T(\boldsymbol{x}+\boldsymbol{u}), R) + \gamma C(\boldsymbol{u}) \right\} \tag{6}$$

which is comprised of three building blocks: a data fidelity term with similarity measure D, a regularization term S and a control term C. The emphasis of this Chapter is how to choose C. To do this for a concrete model, we now specify our choice of all three terms.

3.1 Data Fitting

We consider a similarity measure based on the gradient information [43]. This measure is motivated by the standard **NGF** [22, 32] and it primarily explores the potential of normalized gradients beyond its standard form. We shall consider normalized gradients fitting combined with a measure based on the triangular similarity inequality. More precisely, we consider the following fitting term

$$D(T(\boldsymbol{x}+\boldsymbol{u}), R) = D^{GF}(\boldsymbol{u}) + \alpha D^{TM}(\boldsymbol{u}) \tag{7}$$

where GF stands for 'gradient filed difference' and TM for 'Triangular Measure' with

$$D^{GF}(\boldsymbol{u}) = \int_{\Omega} |\nabla_n T(\boldsymbol{x}+\boldsymbol{u}) - \nabla_n R|^2 \mathrm{d}\boldsymbol{x},$$

$$D^{TM}(\boldsymbol{u}) = \int_{\Omega} (|\nabla T(\boldsymbol{x}+\boldsymbol{u})| + |\nabla R| - |\nabla T(\boldsymbol{x}+\boldsymbol{u}) + \nabla R|)^2 \mathrm{d}\boldsymbol{x}.$$

3.2   Regularization

A regularizer controls the smoothness. Our primary choice for smoothness control is the diffusion model [15] which uses first-order derivatives and promotes smoothness. As affine linear transformations are not included in the kernel of the H^1-regularizer, we desire a regularizer which can penalize such transformation. As such, we add the regularizer based on second-order derivatives (LLT) to the model which allows to remove the need of any pre-registration step of affine transformations. The second-order derivatives allows also getting smooth transformations [52]. Our adopted regularizer is given by

$$S(\boldsymbol{u}) = \frac{\beta_1}{2} S_1(\boldsymbol{u}) + \frac{\beta_2}{2} S_2(\boldsymbol{u}) \tag{8}$$

where

$$S_1(\boldsymbol{u}) = \int_{\Omega} |\nabla \boldsymbol{u}|^2 \mathrm{d}\boldsymbol{x}, \qquad S_2(\boldsymbol{u}) = \int_{\Omega} |\nabla^2 \boldsymbol{u}|^2 \mathrm{d}\boldsymbol{x}.$$

3.3   Invertibility

A diffeomorphic map ensures local invertibility of the map and this is achievable by a control term C that imposes the constraint $\det(J_\varphi) > 0$ at any $\boldsymbol{x} \in \Omega$. This latter idea is much used in the literature with somewhat limited success because either strong assumptions on T, R or compromised fidelity error are required; see tests and remarks from [48]. Here, instead of controlling $\det(J_\varphi)$ directly, we control the Beltrami coefficient [48] in getting a diffeomorphic map and propose the use of

$$C(\boldsymbol{u}) = \int_{\Omega} \phi(|\mu(\boldsymbol{u})|^2) \mathrm{d}\boldsymbol{x}, \tag{9}$$

where $\phi(v) = \frac{v^2}{(v-1)^2}$ and $|\mu(\boldsymbol{u})|^2 = \frac{(\partial_{x_1} u_1 - \partial_{x_2} u_2)^2 + (\partial_{x_2} u_1 + \partial_{x_1} u_2)^2}{(\partial_{x_1} u_1 + \partial_{x_2} u_2 + 2)^2 + (\partial_{x_2} u_1 - \partial_{x_1} u_2)^2}$.

One notes that our choice of the first two terms S, D for (6) is quite common while the third term [48] is relatively new to readers. This is the key idea of this Chapter: an old, non-diffeomorphic, variational model of form (2) can be converted to a diffeomorphic model by adding a control term such as C from (9). This can be done in 2D and also in 3D following our recent work. It should be remarked that model (6) is non-convex so its solutions are not unique (as true for all registration models). However we can show that the model admits at least one solution in the space $W^{2,2}(\Omega)$, following the idea of [49].

4 The Solution Algorithm

Here, we choose first-discretize-then-optimize method, namely directly discretize the variational model to get a discrete optimization problem and then use optimization methods to solve this resulting optimization problem. In this section we focus on a Gauss-Newton (G-N) method and in the next section we briefly introduce another alternating iteration method just before numerical results are shown.

4.1 Discretization

In the implementation, we employ the nodal grid and define a spatial partition

$$\Omega_h^n = \{\boldsymbol{x}^{i,j} \in \Omega | \boldsymbol{x}^{i,j} = (x_1^i, x_2^j) = (ih, jh), 0 \le i \le n, 0 \le j \le n\},$$

where $h = \frac{1}{n}$ and the discrete domain consists of n^2 cells of size $h \times h$. We discretize the displacement field $\boldsymbol{u}$ on the nodal grid, namely $\boldsymbol{u}^{i,j} = (u_1^{i,j}, u_2^{i,j}) = (u_1(x_1^i, x_2^j), u_2(x_1^i, x_2^j))$. By lexicographical ordering, we reshape four matrices to two long vectors of dimension $\mathbb{R}^{2(n+1)^2 \times 1}$

$$X = (x_1^0, x_1^1, ..., x_1^n, \ldots, x_1^0, x_1^1, ...x_1^n, x_2^0, x_2^0, ..., x_2^0, \ldots, x_2^n, x_2^n, ...x_2^n)^T,$$
$$U = (u_1^{0,0}, ..., u_1^{n,0}, \ldots, u_1^{0,n}, ..., u_1^{n,n}, u_2^{0,0}, ..., u_2^{n,0}, \ldots, u_2^{0,n}, ..., u_2^{n,n})^T.$$

4.1.1 Discretization of Fitting Term

Firstly, set $\mathbf{R} = \mathbf{R}(PX) \in \mathbb{R}^{n^2 \times 1}$ as the discretized reference image and $\mathbf{T}(PX + PU) \in \mathbb{R}^{n^2 \times 1}$ as the discretized deformed template image, where $P \in \mathbb{R}^{2n^2 \times 2(n+1)^2}$ is an average matrix from the nodal grid to the cell-centered grid. In order to discretize ∇T and ∇R, we introduce two discrete operators: $D_1 = I_n \otimes \partial_h^1$ and $D_2 = \partial_h^1 \otimes I_n$, where

$$\partial_h^1 = \frac{1}{2h} \begin{bmatrix} -1 & 1 & & & \\ -1 & 0 & 1 & & \\ & ... & ... & ... & \\ & & -1 & 0 & 1 \\ & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

Hence, the discretized ∇T and ∇R are $[D_1\mathbf{T}, D_2\mathbf{T}]$ and $[D_1\mathbf{R}, D_2\mathbf{R}]$ respectively. Set LT $= (\sum_{i=1}^2 D_i\mathbf{T} \odot D_i\mathbf{T} + \epsilon)^{\cdot 1/2}$, LR $= (\sum_{i=1}^2 D_i\mathbf{R} \odot D_i\mathbf{R} + \epsilon)^{\cdot 1/2}$ and LTR $= (\sum_{i=1}^2 D_i(\mathbf{T} + \mathbf{R}) \odot D_i(\mathbf{T} + \mathbf{R}) + \epsilon)^{\cdot 1/2}$, where $\odot$ indicates componentwise product and $(\cdot)^{\cdot 1/2}$ indicates the component-wise square root.

Then for $D^{GF}(\boldsymbol{u})$ and $D^{TM}(\boldsymbol{u})$, we have the following discretizations:

$$D^{GF}(\boldsymbol{u}) \approx h^2 \, p_1^T \, p_1, \quad D^{TSM}(\boldsymbol{u}) \approx h^2 \, p_2^T \, p_2, \tag{10}$$

where (using ./ to indicate the component-wise division)

$$p_1 = [D_1\mathbf{T}./\mathrm{LT} - D_1\mathbf{R}./\mathrm{LR}; \; D_2\mathbf{T}./\mathrm{LT} - D_2\mathbf{R}./\mathrm{LR}]$$
$$p_2 = \mathrm{LT} + \mathrm{LR} - \mathrm{LTR}.$$

4.1.2 Discretization of Regularization Term

The first-order regularization term can be discretized into the following form:

$$S_1(\boldsymbol{u}) \approx h^2 \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{l=1}^{2} \left(\frac{u_l^{i+1,j} - u_l^{i,j}}{h}\right)^2 + \left(\frac{u_l^{i,j+1} - u_l^{i,j}}{h}\right)^2 \tag{11}$$

by using the forward difference and mid-point rule.

Define $B_1 = I_{n+1} \otimes \partial_h^2 \in \mathbb{R}^{(n+1)^2 \times (n+1)^2}$, $C_1 = \partial_h^2 \otimes I_{n+1} \in \mathbb{R}^{(n+1)^2 \times (n+1)^2}$,

$$\partial_h^2 = \frac{1}{h} \begin{bmatrix} -1 & 1 & & \\ \cdots & \cdots & \cdots & \\ & & -1 & 1 \\ & & & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}, \quad A_1 = \begin{bmatrix} B_1 & 0 \\ C_1 & 0 \\ 0 & B_1 \\ 0 & C_1 \end{bmatrix} \in \mathbb{R}^{4(n+1)^2 \times 2(n+1)^2},$$

where $\otimes$ denotes the Kronecker product. Then (11) can be rewritten into the following form (noting $U \in \mathbb{R}^{2(n+1)^2 \times 1}$)

$$S_1(\boldsymbol{u}) \approx h^2 U^T A_1^T A_1 U. \tag{12}$$

The second-order regularization term can be discretized into the following:

$$S_2(\boldsymbol{u}) \approx h^2 \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{l=1}^{2} \left(\frac{u_l^{i+1,j} - 2u_l^{i,j} + u_l^{i-1,j}}{h^2}\right)^2 + \left(\frac{u_l^{i,j+1} - 2u_l^{i,j} + u_l^{i,j-1}}{h^2}\right)^2$$
$$+ 2h^2 \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{l=1}^{2} \left(\frac{u_l^{i,j} - u_l^{i+1,j} - u_l^{i,j+1} + u_l^{i+1,j+1}}{h^2}\right)^2 \tag{13}$$

by using the central difference, mid-point rule and Neumann boundary conditions $(l = 1, 2)$: $u_l^{i,0} = u_l^{i,-1}$, $u_l^{i,n} = u_l^{i,n+1}$, $u_l^{0,j} = u_l^{-1,j}$, $u_l^{n,j} = u_l^{n+1,j}$.

Further define $B_{21} = I_2 \otimes (I_{n+1} \otimes \partial_h^3)$, $B_{22} = I_2 \otimes (\partial_h^3 \otimes I_{n+1})$, $C_2 = I_2 \otimes (E \otimes E)$, $\tau_1 = (n+1) \times (n+1)$, $\tau_2 = n \times (n+1)$, where

$$\partial_h^3 = \frac{1}{h^2} \begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \cdots & \cdots & \cdots & \\ & & 1 & -2 & 1 \\ & & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{\tau_1}, \quad E = \frac{1}{h} \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \cdots & \cdots & \cdots \\ & & & -1 & 1 \\ & & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{\tau_2}.$$

Then (13) can be rewritten into the following form

$$S_2(\boldsymbol{u}) \approx h^2 U^T A_2 U, \quad A_2 = B_{21}^T B_{21} + B_{22}^T B_{22} + 2C_2^T C_2. \tag{14}$$

4.1.3 Discretization of Control Term

Note that $\phi(|\mu(\boldsymbol{u})|^2)$ involves only first order derivatives and all $\boldsymbol{u}^{i,j}$ are available at vertex pixels. Thus it is convenient first to obtain approximations at all cell centers (e.g. at V_5 in Fig. 1) and second to use local linear elements to facilitate first order derivatives. We shall divide each cell (Fig. 1) into 4 triangles. In each triangle, we construct two linear interpolation functions to approximate the u_1 and u_2. Consequently, all partial derivatives are locally constants or $\phi(|\mu(\boldsymbol{u})|^2)$ is constant in each triangle.

Set $\quad \mathbf{L}^{i,j,k}(\boldsymbol{x}) = (L_1^{i,j,k}(\boldsymbol{x}), L_2^{i,j,k}(\boldsymbol{x})) = (a_1^{i,j,k}x_1 + a_2^{i,j,k}x_2 + a_3^{i,j,k}, a_4^{i,j,k}x_1 + a_5^{i,j,k}x_2 + a_6^{i,j,k})$, which is the linear interpolation for $\boldsymbol{u}$ in the $\Omega_{i,j,k}$. Note that $\partial_{x_1}L_1^{i,j,k} = a_1^{i,j,k}$, $\partial_{x_2}L_1^{i,j,k} = a_2^{i,j,k}$, $\partial_{x_1}L_2^{i,j,k} = a_4^{i,j,k}$ and $\partial_{x_2}L_2^{i,j,k} = a_5^{i,j,k}$. Then according to the partition in Fig. 1, we have

Fig. 1 Partition of a cell, nodal point □ and center point ∘. △$V_1 V_2 V_5$ is $\Omega_{i,j,k}$

$$C(\boldsymbol{u}) = \int_\Omega \phi(|\mu(\boldsymbol{u})|^2)\mathrm{d}\boldsymbol{x}$$

$$\approx \frac{h^2}{4} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^4 \phi\Big(\frac{(a_1^{i,j,k} - a_5^{i,j,k})^2 + (a_2^{i,j,k} + a_4^{i,j,k})^2}{(a_1^{i,j,k} + a_5^{i,j,k} + 2)^2 + (a_2^{i,j,k} - a_4^{i,j,k})^2}\Big). \tag{15}$$

To simplify (15), define 3 vectors $\mathbf{r}(U), \mathbf{r}^1(U), \mathbf{r}^2(U) \in \mathbb{R}^{4n^2}$ by $\mathbf{r}(U)_\ell = \mathbf{r}^1(U)_\ell \mathbf{r}^2(U)_\ell$, $\mathbf{r}^1(U)_\ell = (a_1^{i,j,k} - a_5^{i,j,k})^2 + (a_2^{i,j,k} + a_4^{i,j,k})^2$, $\mathbf{r}^2(U)_\ell = 1\big/[(a_1^{i,j,k} + a_5^{i,j,k} + 2)^2 + (a_2^{i,j,k} - a_4^{i,j,k})^2]$ where $\ell = (k-1)n^2 + (j-1)n + i \in [1, 4n^2]$.

Hence, (15) becomes

$$C(\boldsymbol{u}) \approx \frac{h^2}{4}\phi(\mathbf{r}(U))e^T \tag{16}$$

where $\phi(\mathbf{r}(U)) = (\phi(\mathbf{r}(U)_1), ..., \phi(\mathbf{r}(U)_{4n^2}))$ denotes the pixel-wise discretization of u_1, u_2 at all cell centers, and $e = (1, ..., 1) \in \mathbb{R}^{4n^2}$.

Finally, combining the above three parts (10), (12), (14) and (16), we get the discretization formulation for model (6):

$$\min_U J(U) := h^2 p_1^T p_1 + \alpha h^2 p_2^T p_2 + \frac{\beta_1 h^2}{2} U^T A_1^T A_1 U + \frac{\beta_2 h^2}{2} U^T A_2 U + \frac{\gamma h^2}{4}\phi(\mathbf{r}(U))e^T. \tag{17}$$

Remark 1 According to the definition of ϕ and $\mathbf{r}(U)_\ell \geq 0$, each component of $\phi(\mathbf{r}(U))$ is non-negative and differentiable.

4.2 Optimization Method for the Discretized Problem (17)

In the numerical implementation, we choose a line search method to solve the resulting unconstrained optimization problem (17). Here, the basic iterative scheme is

$$U^{i+1} = U^i + \theta\delta U^i, \tag{18}$$

where δU^i is the search direction and θ is the step length. In order to guarantee a descent search direction, we employ a Gauss-Newton method as the standard Newton method does not generate a descent direction because our exact Hessian is non-definite.

4.2.1 Gradient and Approximated Hessian of (17)

Firstly, we consider computing the gradient and approximated Hessian of the discretized fitting term $h^2 p_1^T p_1 + \alpha h^2 p_2^T p_2$. Its gradient and approximated Hessian are respectively:

$$\begin{cases} d_1 = 2h^2 P^T (\mathrm{d}p_1^T p_1 + \alpha \mathrm{d}p_2^T p_2) \in \mathbb{R}^{2(n+1)^2 \times 1}, \\ \hat{H}_1 = h^2 P^T (\mathrm{d}p_1^T \mathrm{d}p_1 + \alpha \mathrm{d}p_2^T \mathrm{d}p_2) P \in \mathbb{R}^{2(n+1)^2 \times 2(n+1)^2}. \end{cases} \tag{19}$$

where $\mathrm{d}p_1 = [\Lambda D_1 - \mathrm{diag}(D_1 \mathbf{T}./t)\Gamma; \Lambda D_2 - \mathrm{diag}(D_2 \mathbf{T}./t)\Gamma]$, $\mathrm{d}p_2 = \sum_{i=1}^{2}$ $\mathrm{diag}(D_i \mathbf{T}./\mathrm{LT} - D_i (\mathbf{T} + \mathbf{R})./\mathrm{LTR})D_i$, $\Lambda = \mathrm{diag}(1./\mathrm{LT})$, $t = \mathrm{LT}^{.3}$, $\Gamma = \sum_{i=1}^{2}$ $\mathrm{diag}(D_i \mathbf{T})D_i$ and $\mathrm{diag}(v)$ is a diagonal matrix with v on its main diagonal.

Remark 2 Evaluating the deformed template image $\mathbf{T}$ must involve interpolation because $PX + PU$ are not in general pixel points. Here in our implementation, we choose B-splines for the interpolation.

For the discretized diffusion regularizer $\frac{\beta_1 h^2}{2} U^T A_1^T A_1 U + \frac{\beta_2 h^2}{2} U^T A_2 U$, its gradient and Hessian are respectively

$$\begin{cases} d_2 = h^2 (\beta_1 A_1^T A_1 + \beta_2 A_2) U \in \mathbb{R}^{2(n+1)^2 \times 1}, \\ H_2 = h^2 (\beta_1 A_1^T A_1 + \beta_2 A_2) \in \mathbb{R}^{2(n+1)^2 \times 2(n+1)^2}. \end{cases} \tag{20}$$

Finally, for the discretized Beltrami term $\frac{\beta h^2}{4} \phi(\mathbf{r}(U))e^T$, the gradient and approximated Hessian are as follows:

$$\begin{cases} d_3 = \frac{\beta h^2}{4} \mathrm{d}\mathbf{r}^T \mathrm{d}\phi(\mathbf{r}) \in \mathbb{R}^{2(n+1)^2 \times 1}, \\ \hat{H}_3 = \frac{\beta h^2}{4} \mathrm{d}\mathbf{r}^T \mathrm{d}^2\phi(\mathbf{r})\mathrm{d}\mathbf{r}. \end{cases} \tag{21}$$

where $\mathrm{d}\phi(\mathbf{r}) = (\phi'(\mathbf{r}_1), ..., \phi'(\mathbf{r}_{4n^2}))^T$ is the vector of derivatives of ϕ at all cell centers,

$$\begin{cases} \mathrm{d}\mathbf{r} = \mathrm{diag}(\mathbf{r}^1)\mathrm{d}\mathbf{r}^2 + \mathrm{diag}(\mathbf{r}^2)\mathrm{d}\mathbf{r}^1, \\ \mathrm{d}\mathbf{r}^1 = 2\mathrm{diag}(A_{31}U)A_{31} + 2\mathrm{diag}(A_{32}U)A_{32}, \\ \mathrm{d}\mathbf{r}^2 = -\mathrm{diag}(\mathbf{r}^2 \odot \mathbf{r}^2)[2\mathrm{diag}(A_{33}U + 2)A_{33} + 2\mathrm{diag}(A_{34}U)A_{34}], \end{cases} \tag{22}$$

$\odot$ denotes a Hadamard product, $\mathrm{d}\mathbf{r}$, $\mathrm{d}\mathbf{r}^1$, $\mathrm{d}\mathbf{r}^2$ are the Jacobian of $\mathbf{r}$, $\mathbf{r}^1$, $\mathbf{r}^2$ with respect to U respectively, $[\mathrm{d}\phi(\mathbf{r})]_\ell$ is the ℓth component of $\mathrm{d}\phi(\mathbf{r})$ and $\mathrm{d}^2\phi(\mathbf{r})$ is the Hessian of ϕ with respect to $\mathbf{r}$, which is a diagonal matrix whose ith diagonal element is $\phi''(\mathbf{r}_i)$, $1 \le i \le 4n^2$. More details about $\mathbf{r}^1$, $\mathbf{r}^2$, A_{31}, A_{32}, A_{33} and A_{34} are shown in Appendix 1.

Therefore, combining the above results for 3 terms, we can obtain the gradient

$$d_J = d_1 + d_2 + d_3 \tag{23}$$

and the approximated Hessian of (17):

$$H = \hat{H}_1 + H_2 + \hat{H}_3. \tag{24}$$

4.2.2 Search Direction

With the above approximated Hessian (24), in each outer (nonlinear) iteration, we solve the Gauss-Newton system

$$H\delta U = -d_J \tag{25}$$

to obtain the search direction δU for (17). Because H is symmetric positive semi-definite, in our implementation, we choose MINRES with diagonal preconditioning as the numerical solver [4, 36].

4.2.3 Step Length

Here, we choose a popular inexact line search condition, *Armijo condition*, which determines a step length θ that satisfies the following sufficient decrease condition:

$$\mathcal{J}(U + \theta\delta U) < \mathcal{J}(U) + \theta\eta d_{\mathcal{J}}^T \delta U. \tag{26}$$

Here, we set $\eta = 10^{-4}$ and use the backtracking approach to find a suitable θ. In addition, we need to check that $\mathbf{r}(U)$ is smaller than 1 which is the norm of the discretized Beltrami coefficient. For more details, please refer to [25, 34, 41].

4.2.4 Stopping Criteria

In the implementation, we choose the stopping criteria used in [33]:

(1.a) $\|J(U^{i+1}) - J(U^i)\| \leq \tau_J(1 + \|J(U^0)\|)$,
(1.b) $\|U^{i+1} - U^i\| \leq \tau_W(1 + \|X + U^0\|)$,
(1.c) $\|d_J\| \leq \tau_G(1 + \|J(U^0)\|)$,
  (2) $\|d_J\| \leq \text{eps}$,
  (3) $i \geq \text{MaxIter}$.

Here, eps is the machine precision and MaxIter is the maximal number of outer iterations. We set $\tau_J = 10^{-3}, \tau_W = 10^{-2}$ and $\tau_G = 10^{-2}$. If any one of (1) (2) and (3) is satisfied, the iterations are terminated. Hence, a Gauss-Newton numerical scheme with Armijo line search can be developed and summarized in Algorithm 1.

Algorithm 1 Gauss-Newton scheme by using Armijo line search for Image Registration: $U \leftarrow \text{GNAIR}(\alpha, \beta_1, \beta_2, \gamma, U^0, T, R)$

Step 1: Set $i = 0$ at the solution point $U^i = U^0$.
Step 2: For (17), compute the energy functional $J(U^i)$, its gradient d_J^i and
   the approximated Hessian H^i by (24).
while "none of the listed 3 stopping criteria are satisfied" **do**
 Solve the Gauss-Newton equation: $H^i \delta U^i = -d_J^i$;
 Use Line Search to find step length θ;
 $U^{i+1} = U^i + \theta \delta U^i$;
 $i = i + 1$;
 Compute $J(U^i)$, d_J^i and H^i;
end while

4.2.5 Multi-level Strategy

A multi-level strategy is a standard technique in image registration. In the multi-level strategy, we firstly coarsen the template T and the reference R by L levels. Then we can obtain U_1 by solving our model (6) on the coarsest level. In order to give a good initial guess for the finer level, we adopt an interpolation operator on U_1 to obtain U_2^0 as the initial guess for the next level. We repeat this process and can get the final registration on the finest level. The most important advantage of the multi-level strategy is that it can save computation time because of less variables on the coarser level than on the fine level. In addition, it can help to avoid trapping into a local minimum.

4.2.6 Convergence Result

Our above described Algorithm 1 will converge to a stationary point of our new model. Details are shown in Theorem 1 of Appendix 2 below.

5 Numerical Results

In this section, we will show some numerical results to illustrate the performances of our proposed model (6) using Gauss-Newton method called **GNR**. We compare with the standard **NGF** [32] and the Augmented Lagrangian approach for solving a similar model [43] called **ALMR**, which uses the same regularization and fitting terms. However, the local invertibility of the map is guaranteed by imposing an inequality constraint on the model. For more details about the augmented Lagrangian method, we refer to [3, 38, 42] and the reference therein.

ALMR. Alternating iteration is another popular method which might be applied to (6). However, below, we shall consider it for a related model [43] that uses a constrained optimization (different from (6)):

$$
\begin{cases}
\min\limits_{\boldsymbol{u}\in\mathcal{H}}\{\mathcal{J}_1(\boldsymbol{u}) = S(\boldsymbol{u}) + \dfrac{\lambda}{2}D^{GF}(\boldsymbol{u}) + \dfrac{\lambda}{2}D^{TM}(\boldsymbol{u})\}, \\
\text{w.r.t} \quad \mathcal{C}_\epsilon(\boldsymbol{u}) = \det(I + \nabla\boldsymbol{u}) \geq \epsilon,
\end{cases}
\tag{27}
$$

where imposing the constraint is a competing way of ensuring a diffeomorphic transformation.

To reformulate (27), introducing variables K, $\mathbf{p}$ and $\mathbf{n}$, we solve the following constrained minimization problem:

$$
\begin{cases}
\min\limits_{\boldsymbol{u},K,\mathbf{p},\mathbf{n}}\{S(\boldsymbol{u}) + \dfrac{\lambda}{2}\displaystyle\int_\Omega (\mathbf{n} - \nabla_n R)^2 \mathrm{d}\boldsymbol{x} + \dfrac{\lambda}{2}\displaystyle\int_\Omega (|\mathbf{p}| + |\nabla R| - |\mathbf{m}|)^2\, \mathrm{d}\boldsymbol{x}\}, \\
\text{w.r.t} \quad K = T(\boldsymbol{x}+\boldsymbol{u}), \ \ \mathbf{p} = \nabla K, \ \ |\mathbf{p}|\mathbf{n} = \mathbf{p}, \ \ \mathbf{m} = \mathbf{p} + \nabla R, \ \ \mathcal{C} > 0.
\end{cases}
\tag{28}
$$

Then, the augmented Lagrangian functional corresponding to the constrained optimization problem (28) is defined as follows:

$$
\begin{aligned}
&\mathcal{L}_1(\boldsymbol{u}, K, \mathbf{p}, \mathbf{n}, \mathbf{m}, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) \\
&= S(\boldsymbol{u}) + \frac{\lambda}{2}\int_\Omega (\mathbf{n} - \nabla_n R)^2 \mathrm{d}\boldsymbol{x} + \frac{\lambda}{2}\int_\Omega (|\mathbf{p}| + |\nabla R| - |\mathbf{m}|)^2\, \mathrm{d}\boldsymbol{x} \\
&\quad + \frac{r_2}{2}\int_\Omega (\mathbf{p} - \nabla K)^2 \mathrm{d}\boldsymbol{x} + \frac{r_3}{2}\int_\Omega (\mathbf{p} - |\mathbf{p}|\mathbf{n})^2 \mathrm{d}\boldsymbol{x} + \frac{r_4}{2}\int_\Omega (\mathbf{p} + \nabla R - \mathbf{m})^2 \mathrm{d}\boldsymbol{x} \\
&\quad + \int_\Omega (T(\boldsymbol{x}+\boldsymbol{u}) - K)\lambda_1 \mathrm{d}\boldsymbol{x} + \int_\Omega (\mathbf{p} - \nabla K)\cdot\lambda_2 \mathrm{d}\boldsymbol{x} + \int_\Omega (\mathbf{p} - |\mathbf{p}|\mathbf{n})\cdot\lambda_3 \mathrm{d}\boldsymbol{x} \\
&\quad + \int_\Omega (\mathbf{p} + \nabla R - \mathbf{m})\cdot\lambda_4\, \mathrm{d}\boldsymbol{x} + \frac{r_1}{2}\int_\Omega (T(\boldsymbol{x}+\boldsymbol{u}) - K)^2 \mathrm{d}\boldsymbol{x} + \frac{1}{2\sigma}\int_\Omega \mathcal{C}_s(\boldsymbol{u}, \lambda_5)\, \mathrm{d}\boldsymbol{x},
\end{aligned}
\tag{29}
$$

where

$$
\mathcal{C}_s(\boldsymbol{u}, \lambda_5) = [\min\{0, \sigma(\mathcal{C}(\boldsymbol{u}) - \epsilon) - \lambda_5)\}]^2 - \lambda_5^2,
\tag{30}
$$

$\epsilon > 0$ is a small parameter, $\sigma > 0$ and $\lambda_i\,(i = 1, \ldots, 5)$ are the Lagrange multipliers. The augmented Lagrangian algorithm is shown in Algorithm 2.

Algorithm 2 Augmented Lagrangian method

---

1. Initialization: $\boldsymbol{u}^0$, K^0, $\mathbf{p}^0$, $\mathbf{n}^0$, $\mathbf{m}^0$ and λ_1^0, λ_2^0, λ_3^0, λ_4^0 and λ_5^0.
2. Iterate for $k = 1, 2, \ldots$ until a required tolerance:
   — compute an approximate minimizers $\mathbf{u}^{k+1}$, K^{k+1}, $\mathbf{p}^{k+1}$, $\mathbf{n}^{k+1}$ and $\mathbf{m}^{k+1}$ of the augmented Lagrangian functional with the fixed Lagrange multipliers λ_1^k, λ_2^k, λ_3^k, λ_4^k and λ_5^k:

$$\left[\boldsymbol{u}^{k+1}, K^{k+1}, \mathbf{p}^{k+1}, \mathbf{n}^{k+1}, \mathbf{m}^{k+1}\right] =$$
$$\underset{\boldsymbol{u}, K, \mathbf{p}, \mathbf{n}}{\arg \min} \mathcal{L}_1(\boldsymbol{u}, K, \mathbf{p}, \mathbf{n}, \mathbf{m}, \lambda_1^k, \lambda_2^k, \lambda_3^k, \lambda_4^k, \lambda_5^k). \tag{31}$$

   — Update Lagrange multipliers

$$\lambda_1^{k+1} = \lambda_1^k + r_1(T(\boldsymbol{x} + \boldsymbol{u}^{k+1}) - K^{k+1}), \tag{32}$$
$$\lambda_2^{k+1} = \lambda_2^k + r_2(\mathbf{p}^{k+1} - \nabla K^{k+1}), \tag{33}$$
$$\lambda_3^{k+1} = \lambda_3^k + r_3(\mathbf{p}^{k+1} - |\mathbf{p}^{k+1}|\mathbf{n}^{k+1}), \tag{34}$$
$$\lambda_4^{k+1} = \lambda_4^k + r_4(\mathbf{m}^{k+1} - \mathbf{p}^{k+1} - \nabla R), \tag{35}$$
$$\lambda_5^{k+1} = \max\{0, \lambda_5^k - \sigma\mathcal{C}_\epsilon(\boldsymbol{u}^{k+1})\}, \tag{36}$$

In practice, the minimization problem (29) or (31) is decomposed into a number of sub-problems, each of which can be solved quickly. However, the convergence of the augmented Lagrangian iterations for this case is not guaranteed due to the non-convexity of overall registration problem. Currently this is a major weakness of **ALMR** while the convergence of **GNR** (even if a bit slower) can be proved and hence recommended.

In order to reduce the number of parameters to tune, we set $\lambda = 15$, $\beta_1 = 0.005$, $\beta_2 = 0.1 \times \beta_1$ $r_1 = 5$, $r_2 = 10$ and $r_3 = r_4 = 100$ in all numerical experiments unless stated otherwise. We consider $N_{max} = 70$ as the maximum number of iterations for **ALMR** from Algorithm 2 and we stop the iterations before reaching $N_{max} = 70$ if the following stopping criterion

$$\frac{\|\mathbf{p}^k + \nabla R - \mathbf{m}^k\|_{L^1}}{\sqrt{l \times c}} \leq \tau$$

is satisfied for a given tolerance $\tau = 10^{-3}$, where l and c are the numbers of rows and columns in the image

For all compared methods, we set the zero vector as the initial guess U^0. To measure the quality of the registered images, we use the following quantities

$$\text{GFer} = \frac{D^{GF}(\boldsymbol{u})}{D^{GF}(\boldsymbol{u}^0)}, \tag{37}$$

$$\text{NGFer} = \frac{D^{NGF}(\boldsymbol{u})}{D^{NGF}(\boldsymbol{u}^0)}, \tag{38}$$

and

$$\text{MIer} = -D^{MI}(\boldsymbol{u}). \tag{39}$$

The good result means that it can lead to small GFer, small NGFer and large MIer. All the codes are implemented by Matlab R2019b on a PC with 3.4 GHz Intel(R) Core(TM) i5-3570 processor and 12 GB RAM.

5.1 Example 1

In this example, we consider a pair of images displayed in Fig. 2a, b. The resolution is 256×256. In order to choose the parameter easily, in this example, we fix α and set $\alpha = 0.01$.
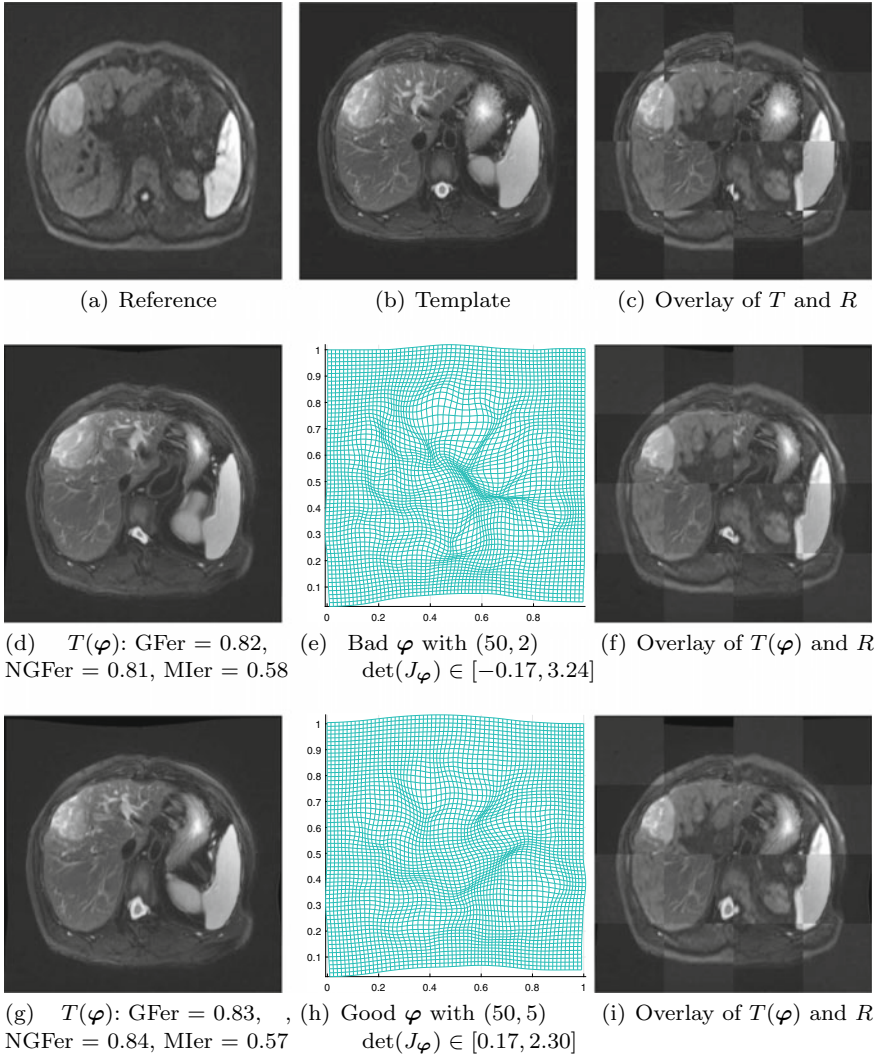
Firstly, we consider the model without Beltrami control term, namely $\gamma = 0$. For the parameters of regularizers, we set two pairs $(\beta_1, \beta_2) = (50, 2)$ and $(\beta_1, \beta_2) = (50, 5)$. The corresponding deformed templates and transformations are shown in Fig. 2d, e, g, h. From Fig. 2f, i, we can find that the deformed templates generated by these two pairs of parameters are visually satisfied. In addition, these two choices give similar measurements: GFer = 0.82, NGFer = 0.81, MIer = 0.58 and GFer = 0.83, NGFer = 0.84, MIer = 0.57 respectively. However, the first choice leads to a transformation containing folding because the minimum of the Jacobian determinant of the transformation is negative but the second choice produces a smooth transformation without folding because the minimum of the Jacobian determinant of the transformation is positive.

Since first and second order regularizers just control the smoothness, in order to overcome this drawback, we keep $(\beta_1, \beta_2) = (50, 2)$ unchanged and choose a suitable γ. Here, we set $\gamma = 10$. Figure 3a, b shows the corresponding deformed template and transformation. From Fig. 3c, the deformed template is similar visually with the previous one without controlling the Beltrami coefficient and the measurements are also similar (GFer = 0.82, NGFer = 0.82 and MIer = 0.57). But the minimum of the Jacobian determinant of the transformation is positive, which illustrates that the transformation is diffeomorphic. In the same figure, we also give the result of **ALMR** model, which shows again from the overlay of $T(\varphi)$ and the reference R that the template image T is well registered to R.
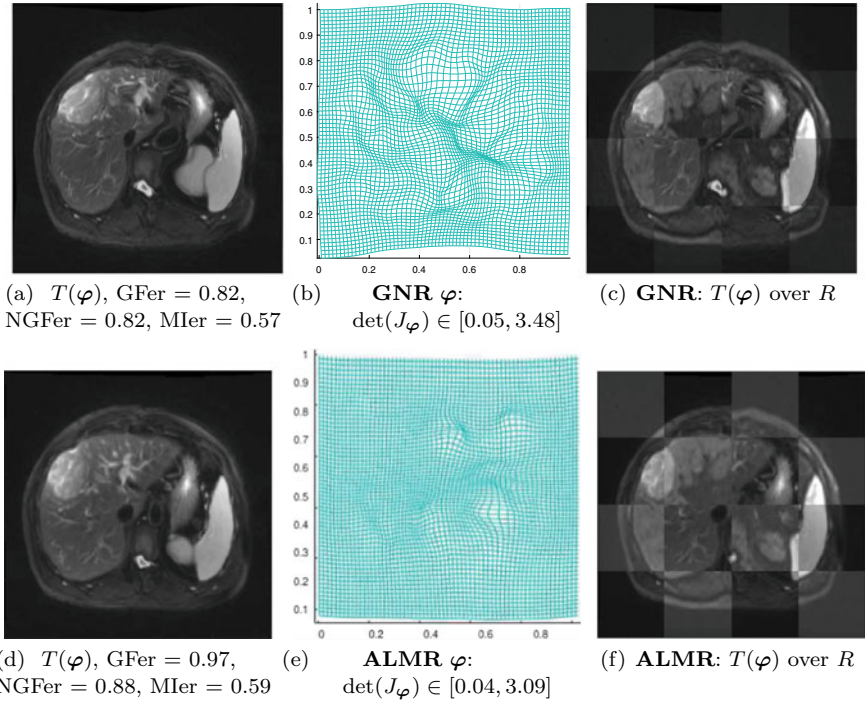
Now, we investigate the sensitivity of γ. From Table 1, we can find that when we fix α, β_1 and β_2 and change γ, GFer, NGFer and MIer are robust and at the same time, the minimum of the Jacobian determinant of the transformations are all positive. This indicates that the Beltrami control term is not sensitive.

In addition, we also investigate the convergence of the algorithm for our model. Here, we force the relative norm of the gradient of the approximated solution to reach 10^{-3} although it only runs several iterations by using the practical stopping criteria. Here, according to Fig. 4, we can find that the algorithm for our model is convergent.

(a) Reference

(b) Template

(c) Overlay of T and R

(d) $T(\varphi)$: GFer = 0.82, NGFer = 0.81, MIer = 0.58

(e) Bad φ with $(50, 2)$ $\det(J_\varphi) \in [-0.17, 3.24]$

(f) Overlay of $T(\varphi)$ and R

(g) $T(\varphi)$: GFer = 0.83, NGFer = 0.84, MIer = 0.57

(h) Good φ with $(50, 5)$ $\det(J_\varphi) \in [0.17, 2.30]$

(i) Overlay of $T(\varphi)$ and R

Fig. 2 Example 1 **without the Beltrami control term**: the first row shows the reference, template and overlay of the reference and template. The second and third rows show the deformed templates and transformations obtained by two pairs of parameters $(\beta_1, \beta_2) = (50, 2)$ and $(\beta_1, \beta_2) = (50, 5)$, respectively. The results are visually similar but the transformations are not both one-to-one. The first choice leads to a mesh with folding because the minimum of the Jacobian determinant of the transformation is negative

(a) $T(\varphi)$, GFer $= 0.82$,
NGFer $= 0.82$, MIer $= 0.57$

(b)   **GNR** φ:
$\det(J_\varphi) \in [0.05, 3.48]$

(c) **GNR**: $T(\varphi)$ over R

(d) $T(\varphi)$, GFer $= 0.97$,
NGFer $= 0.88$, MIer $= 0.59$

(e)   **ALMR** φ:
$\det(J_\varphi) \in [0.04, 3.09]$

(f) **ALMR**: $T(\varphi)$ over R

Fig. 3 Example 1: the deformed template and transformation are generated by $(\beta_1, \beta_2, \gamma) = (50, 2, 10)$. The results are visually satisfied and the transformation is one-to-one. Second row: the deformed template obtained by **ALMR** and its overlay with the reference R

Table 1 Example 1: measurements obtained by using $\alpha = 10^{-2}$, $\beta_1 = 50$ and $\beta_2 = 2$

γ	GFer	NGFer	MIer	min $\det(J_\varphi)$	max $\det(J_\varphi)$
1	0.82	0.82	0.57	0.01	3.14
10	0.82	0.82	0.57	0.05	3.48
100	0.82	0.82	0.57	0.06	3.13
1000	0.82	0.82	0.57	0.21	3.11

Hence, this example illustrates that our new control term can effectively control the transformation and lead to an accurate registration. Meanwhile, the new control term can make this model more robust.

(a) Relative norm of the gradient   (b) Relative norm of the function value
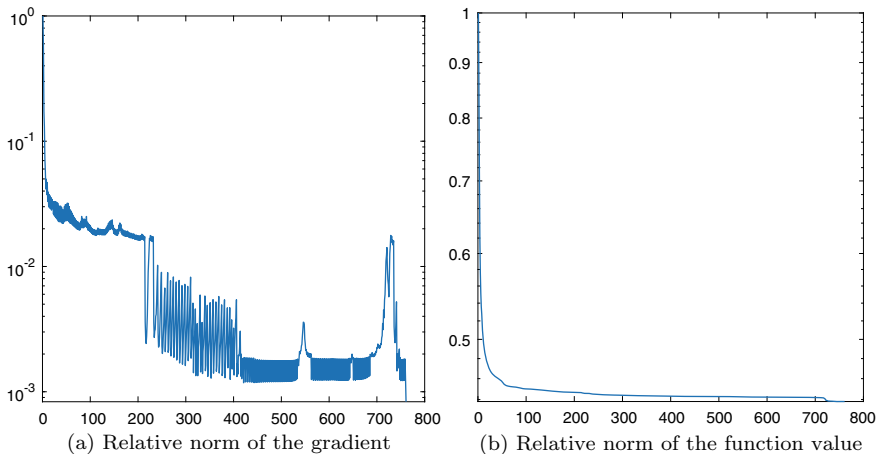
Fig. 4 Example 1: Relative norm of the gradient and relative norm of the function value by the parameter $(\alpha, \beta_1, \beta_2, \gamma) = (0.01, 50, 2, 10)$. Here, we can notice that our algorithm is convergent

5.2 *Example 2*

In this example, we consider another pair of 256×256 images (Fig. 5a, b). Again, in order to reduce the complexity of choosing parameters, we fix $\alpha = 10^{-1}$ in this example.

Firstly, we set $\beta_1 = 50$, $\beta_2 = 10$ and $\gamma = 0$. From Fig. 5d–f, although the deformed template is satisfied visually, we can find that the resulting transformation has folding since the minimum of the Jacobian determinant is negative.

As a comparison, we also test the model of the standard **NGF** [32] with the same first- and second-order regularizer. Here, we test three pairs of (β_1, β_2) and the corresponding results are shown in Fig. 6. We can find that for the fitting term, if we choose **NGF**, it is very hard to choose the suitable parameters to get a good registration, namely, simultaneously get a diffeomorphic transformation and a visually satisfied deformed template. In order to overcome this difficulty, we keep β_1, β_2 unchanged and choose γ as 1, 10 and 100 separately. Figure 7 shows that they can all generate visually satisfied deformed template and diffeomorphic transformations. Specifically, according to Fig. 7, we can see that the measurements obtained by these choices are very similar, which again demonstrates that this model can be more robust through combining the Beltrami control term. We also give the result of **ALMR** model in Fig. 8. We can observe from overlay of the registered and the reference images that all models work fine in producing acceptable registration result.

(a) Reference   (b) Template   (c) Overlay of T and R

(d) $T(\boldsymbol{\varphi})$, GFer $= 0.62$, (e)   **GNR**: $\boldsymbol{\varphi}$,   (f) **GNR**: $T(\boldsymbol{\varphi})$ over R
NGFer $= 0.68$, MIer $= 0.83$ $\det(J_{\boldsymbol{\varphi}}) \in [-0.19, 3.13]$
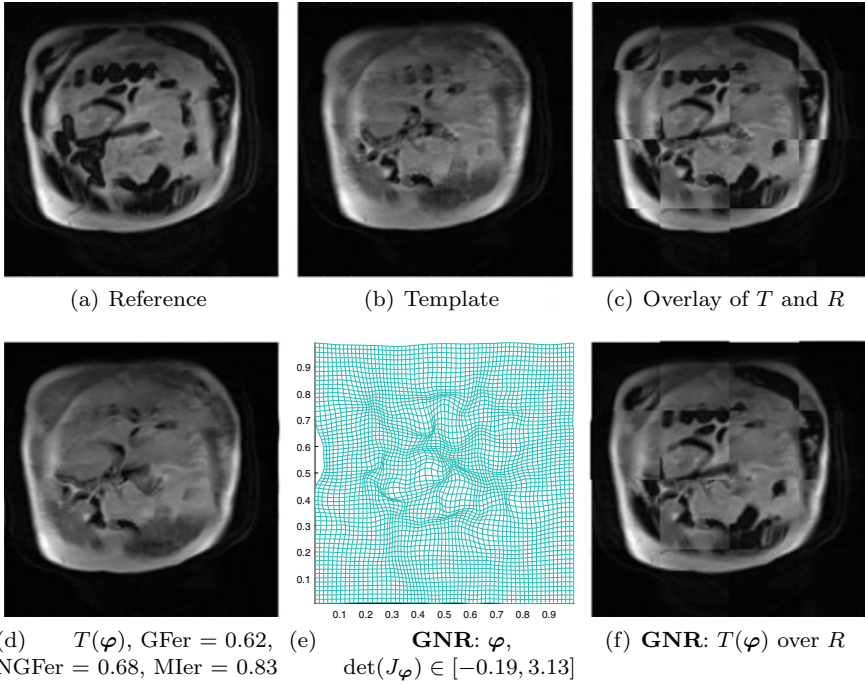
Fig. 5 Example 2 by the new model **GNR** without using the control term C: the resulting transformation is not diffeomorphic although the deformed template is visually satisfied
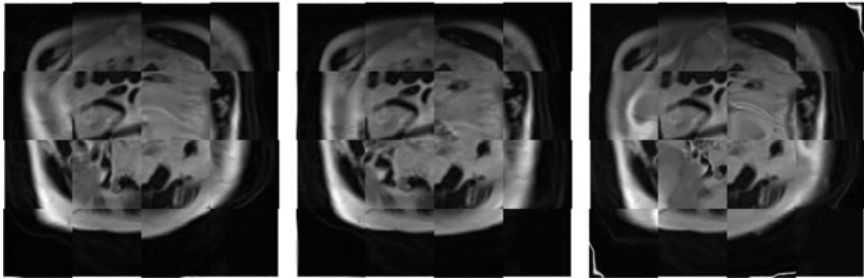
In summary, when the **ALMR**, the **NGF** and the **GNR** work, the latter has the largest MIer similarity (indicating better quality). However, **NGF** (or taking out an extra control term for **ALMR** and **GNR**) can fail to deliver a valid result (with negative $\det \nabla \boldsymbol{y}$) if the parameters are not chosen correctly. Although **ALMR** is completive to **GNR** (and takes less time to converge in practice), only the convergence of **GNR** can be proved. Hence our model **GNR** is robust and can be recommended for multi-modal registration.

6 Conclusions

Image registration is an increasingly important and often challenging image processing task. The quality of the transformation requires suitable control. In this Chapter to improve a multi-modality registration model, we propose a novel term motivated by Beltrami coefficient, which can lead to a diffeomorphic transformation. The advantage of the term lies in no bias imposed on its Jacobian of the transformation's determinant. By employing first-discretize-then-optimize method, we design an effective
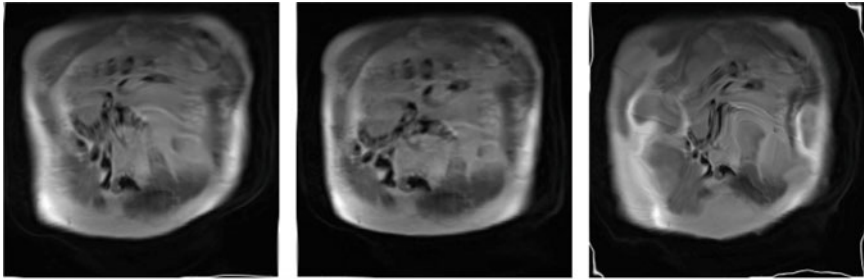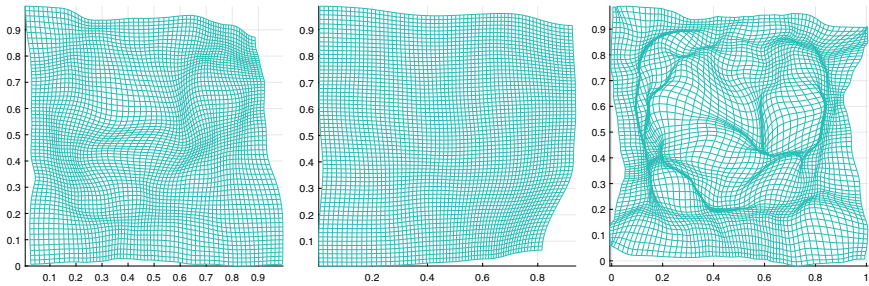
(a) $T(\varphi)$ with GFer = 1.08, NGFer = 0.96, MIer = 0.50 with $(\beta_1, \beta_2) = (0.1, 0.001)$

(b) $T(\varphi)$ with GFer = 1.07, NGFer = 0.96, MIer = 0.53 with $(\beta_1, \beta_2) = (0.01, 0.01)$

(c) $T(\varphi)$: GFer = 1.14, NGFer = 0.99, MIer = 0.50 with $(\beta_1, \beta_2) = (0.01, 10^{-4})$

(d) $T(\varphi)$ over R
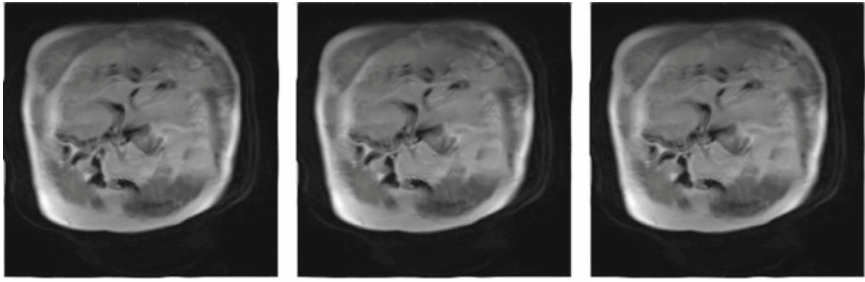
(e) $T(\varphi)$ over R

(f) $T(\varphi)$ over R
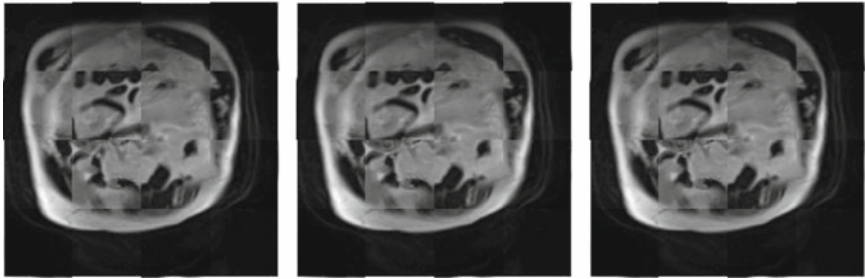
(g) φ
$\det(J_{\varphi}) \in [0.27, 1.94]$

(h) φ
$\det(J_{\varphi}) \in [0.54, 1.19]$

(i) φ
$\det(J_{\varphi}) \in [-0.89, 4.10]$

Fig. 6 Example 2 by the **GNR without** imposing a control term. Each column shows results of a different choice of (β_1, β_2) balancing first- and second-order regularizers: the deformed template, overlay of $T(\varphi)$ and R, and the transformation. Clearly the last column obtains the incorrect φ
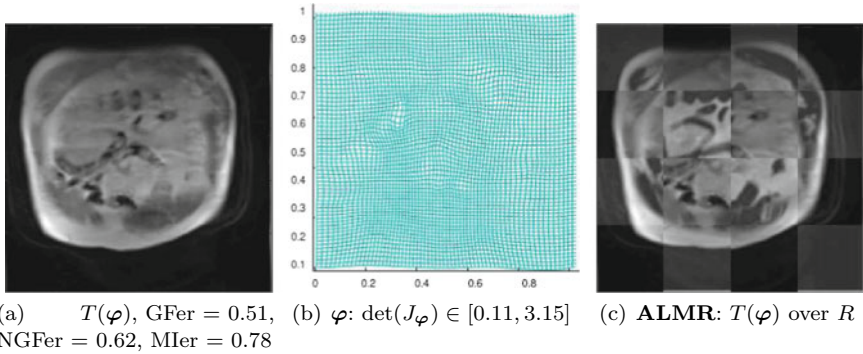
(a) $T(\varphi)$: GFer $= 0.62$, NGFer $= 0.68$, MIer $= 0.83$ with $\gamma = 1$

(b) $T(\varphi)$: GFer $= 0.63$, NGFer $= 0.68$, MIer $= 0.83$ with $\gamma = 10$

(c) $T(\varphi)$: GFer $= 0.63$, NGFer $= 0.68$, MIer $= 0.83$ with $\gamma = 100$

(d) $T(\varphi)$ over R

(e) $T(\varphi)$ over R

(f) $T(\varphi)$ over R
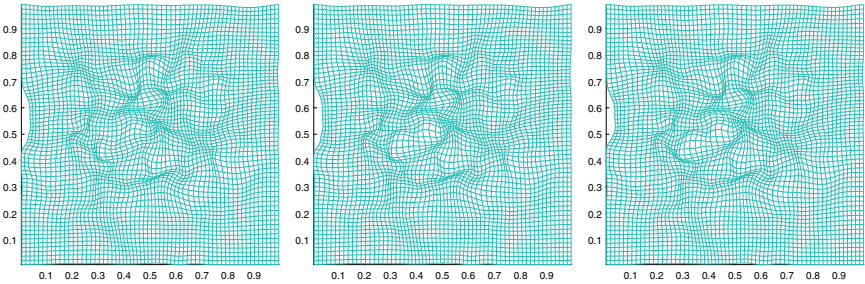
(g) φ: $\det(J_\varphi) \in [0.01, 2.54]$ (h) φ: $\det(J_\varphi) \in [0.01, 3.27]$ (i) φ: $\det(J_\varphi) \in [0.02, 3.19]$

Fig. 7 Example 2 by the new model **GNR**. By using the control term for each choice of γ (by column), the resulting transformation is diffeomorphic and the deformed template is also visually pleasing

solver to solve our proposed model numerically. Experimental tests confirm that our proposed model performs well in multi-modality images registration. In addition, with the help of the Beltrami control term, the proposed model is more robust with respect to the parameters. Future work will investigate extension of this work to a deep learning framework [45].

(a) $T(\varphi)$, GFer $= 0.51$,   (b) φ: $\det(J_\varphi) \in [0.11, 3.15]$ (c) **ALMR**: $T(\varphi)$ over R
NGFer $= 0.62$, MIer $= 0.78$

Fig. 8 Example 2 by **ALMR** model. The deformed template is also visually close to the reference R

Appendix 1—Computation of the Vector r(U)

First of all, denote the 3 vertices of this triangle by $V_1 = x^{1,1}$, $V_2 = x^{2,1}$ and $V_5 = x^{1.5,1.5}$ in Fig. 1. Set $\mathbf{L}(V_1) = (u_1^{1,1}, u_2^{1,1})$, $\mathbf{L}(V_2) = (u_1^{2,1}, u_2^{2,1})$ at the vertex pixels, and $\mathbf{L}(V_5) = (u_1^{1.5,1.5}, u_2^{1.5,1.5})$ at the cell centre (approximated values). Here the linear approximations are $\mathbf{L}(x_1, x_2) = (a_1 x_1 + a_2 x_2 + a_3, a_4 x_1 + a_5 x_2 + a_6)$.

After substituting V_1, V_2 and V_5 into $\mathbf{L}$, we get

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \frac{1}{\det} \begin{pmatrix} x_2^1 - x_2^{1.5} & -x_2^1 + x_2^{1.5} \\ -x_1^2 + x_1^{1.5} & x_1^1 - x_1^{1.5} \end{pmatrix} \begin{pmatrix} u_1^{1,1} - u_1^{1.5,1.5} \\ u_1^{2,1} - u_1^{1.5,1.5} \end{pmatrix}, \tag{40}$$

$$\begin{pmatrix} a_4 \\ a_5 \end{pmatrix} = \frac{1}{\det} \begin{pmatrix} x_2^1 - x_2^{1.5} & -x_2^1 + x_2^{1.5} \\ -x_1^2 + x_1^{1.5} & x_1^1 - x_1^{1.5} \end{pmatrix} \begin{pmatrix} u_2^{1,1} - u_2^{1.5,1.5} \\ u_2^{2,1} - u_2^{1.5,1.5} \end{pmatrix}, \tag{41}$$

where $\det = \begin{vmatrix} x_1^1 - x_1^{1.5} & x_2^1 - x_2^{1.5} \\ x_1^2 - x_1^{1.5} & x_2^1 - x_2^{1.5} \end{vmatrix}$.

According to (40) and (41), we can formulate two matrices $D1 \in \mathbb{R}^{4n^2 \times (n+1)^2}$ and $D2 \in \mathbb{R}^{4n^2 \times (n+1)^2}$ such that

$$A_{31} = [D1, -D2], \quad A_{32} = [D2, D1], \quad A_{33} = [D1, D2], \quad A_{34} = [D2, -D1].$$

Then using the Hadamard product $\odot$, we get a compact form for

$$\begin{cases} \mathbf{r}^1(U) = A_{31}U \odot A_{31}U + A_{32}U \odot A_{32}U, \\ \mathbf{r}^2(U) = 1./((A_{33}U + 2) \odot (A_{33}U + 2) + A_{34}U \odot A_{34}U), \\ \mathbf{r}(U) = \mathbf{r}^1 \odot \mathbf{r}^2 \in \mathbb{R}^{4n^2 \times 1}. \end{cases} \tag{42}$$

Appendix 2—The Global Convergence of Algorithm 1

In order to discuss the global convergence result of Algorithm 1 for the discretized optimization problem (17), we first review two lemmas.

Lemma 1 ([25]) *For the unconstrained optimization problem*

$$\min_U J(U)$$

let an iterative sequence be defined by $U^{i+1} = U^i + \theta \delta U^i$, where $\delta U^i = -(H^i)^{-1} d_J(U^i)$ and θ is obtained by Armijo condition. Assume that three conditions are met: (i). d_J be Lipschitz continuous; (ii). the matrices H^i are SPD (iii). there exist constants $\bar{\kappa}$ and M such that the condition number $\kappa(H^i) \leq \bar{\kappa}$ and the norm $||H^i|| \leq M$ for all i. Then either $J(U^i)$ is unbounded from below or

$$\lim_{i \to \infty} d_J(U^i) = 0 \tag{43}$$

and hence any limit point of the sequence of iterates is a stationary point.

Lemma 2 *Let a matrix be comprised of 3 submatrices $H = H_1 + H_2 + H_3$. If H_1 and H_2 are symmetric positive semi-definite and H_3 is SPD, then H is SPD with $\lambda_{H_3} \leq \lambda_H$, where λ_{H_3} and λ_H are the minimum eigenvalues of H_3 and H separately.*

Proof According to Rayleigh quotient, we can find a vector v such that

$$\lambda_H = \frac{v^T H v}{v^T v}. \tag{44}$$

Then we have

$$\lambda_{H_3} \leq \frac{v^T H_1 v}{v^T v} + \frac{v^T H_2 v}{v^T v} + \frac{v^T H_3 v}{v^T v} = \frac{v^T H v}{v^T v} = \lambda_H, \tag{45}$$

which completes the proof.

In the above discretization leading to (17), we do not need to introduce the boundary condition. However for theory purpose, in the following, we will prove our convergence result under the Dirichlet boundary condition (namely, the boundary is fixed) and this condition is needed to prove the symmetric positive definite (SPD) property of the approximated Hessian. In practical implementation, such a condition is not required as confirmed by experiments.

In addition, define an important set $\mathcal{X} := \{U \mid \mathbf{r}(U)_\ell \leq 1 - \epsilon, 1 \leq \ell \leq 4n^2\}$ for small ϵ. So $U \in \mathcal{X}$ means that the transformation is diffeomorphic. Under the suitable γ, each U^i generated by Algorithm 1 is in the $\mathcal{X}$.

Theorem 1 *Assume that T and R are twice continuously differentiable. For* (17)*, by using Algorithm 1, we obtain*

$$\lim_{i \to \infty} d_J(U^i) = 0 \tag{46}$$

and hence any limit point of the sequence of iterates produced by Algorithm 1 is a stationary point.

Proof It suffices to show that Algorithm 1 satisfies the requirements of Lemma 1. Recall $\mathbf{r}(U)$ and we can see that it is continuous. Here, we use the Dirichlet boundary condition and we can assume that $\|U\|$ is bounded. Then $\mathbf{r}(U)$ is a continuous mapping from a compact set to $\mathbb{R}^{4n^2 \times 1}$ and $\mathbf{r}(U)$ is proper. So for some small $\epsilon > 0$, $\mathcal{X}$ is compact.

Firstly, we show that in $\mathcal{X}$, d_J of (17) is Lipschitz continuous. The term $\phi(\mathbf{r}(U))e^T$ in the (17) is twice continuously differentiable with respect to $U \in \mathcal{X}$. In addition, T and R are twice continuously differentiable. So (17) is twice continuously differentiable with respect to $U \in \mathcal{X}$ and d_J is Lipschitz continuous.

Secondly, we show that in $\mathcal{X}$, $H^i = \hat{H}_1^i + H_2^i + \hat{H}_3^i$ is SPD. By the construction of $\hat{H}_1^i$ and $\hat{H}_3^i$, they are symmetric positive semi-definite. H_2^i is symmetric positive definite under the Dirichlet boundary condition. Consequently, according to Lemma 2, H^i is SPD.

Thirdly, we show that both $\kappa(H^i)$ and $\|H^i\|$ are bounded. We notice that in each iteration, H_2^i is constant and we can set $\|H_2^i\| = M_2$. For $\hat{H}_1^i$, we get its upper bound M_1 because T is twice continuously differentiable and $\mathcal{X}$ is compact. ϕ is also twice continuously differentiable with respect to $U \in \mathcal{X}$, then we have $\|\hat{H}_3^i\| \leq M_3$. Hence, we have

$$\|H^i\| \leq \|\hat{H}_1^i\| + \|H_2^i\| + \|\hat{H}_3^i\| \leq M_1 + M_2 + M_3. \tag{47}$$

So set $M = M_1 + M_2 + M_3$ and $\|H^i\| \leq M$. Set σ as the minimum eigenvalue of H_2^i. According to Lemma 2, the smallest eigenvalue λ_{min} of H^i should be larger than σ. The largest eigenvalue λ_{max} of H^i should be smaller than M due to $\lambda_{max} \leq \|H^i\|$. So the conditional number of H^i is smaller than $\frac{M}{\sigma}$.

Finally, we can find that (17) has lower bound 0. Hence, by applying Lemma 1, we complete the proof.

References

1. A. Angelov, M. Wagner, Multimodal image registration by elastic matching of edge sketches via optimal control. Journal of Industrial & Management Optimization **10**, 567–590 (2014)
2. N. Aronszajn, Theory of reproducing kernels. Transactions of the American mathematical society **68**, 337–404 (1950)
3. E. Bae, J. Shi, X.-C. Tai, Graph cuts for curvature based image denoising. IEEE Transactions on Image Processing **20**, 1199–1210 (2011)

4. R. Barrett, M.W. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. Van der Vorst, Templates for the solution of linear systems: building blocks for iterative methods, vol. 43 (Siam, 1994)

5. A. Berlinet, C. Thomas-Agnan, *Reproducing Kernel Hilbert Spaces in Probability and Statistics* (Springer Science & Business Media, 2011)

6. K. Briechle, U.D. Hanebeck, Template matching using fast normalized cross correlation, in *Optical Pattern Recognition XII*, vol. 4387 (International Society for Optics and Photonics, 2001), pp. 95–102

7. M. Burger, J. Modersitzki, L. Ruthotto, A hyperelastic regularization energy for image registration. SIAM Journal on Scientific Computing **35**, B132–B148 (2013)

8. K. Chen, L. M. Lui, J. Modersitzki, Image and surface registration, in *Learning of Images, Shapes, and Forms*, ed. by R. Kimmel, X.-C. Tai, vol. 20 (Handbook of Numerical Analysis, North Holland - Elsevier, 2019), pp. 579–611

9. Y.M. Chen, J.L. Shi, M. Rao, J.S. Lee, Deformable multi-modal image registration by maximizing renyi's statistical dependence measure. Inverse Problems and Imaging **9**, 79–103 (2015)

10. N. Chumchob, Vectorial total variation-based regularization for variational image registration. IEEE Trans. Image Processing **22**, 4551–4559 (2013)

11. N. Chumchob, K. Chen, Improved variational image registration model and a fast algorithm for its numerical approximation. Numerical Methods for Partial Differential Equations **28**, 1966–1995 (2012)

12. N. Chumchob, K. Chen, C. Brito-Loeza, A fourth-order variational image registration model and its fast multigrid algorithm. Multiscale Modeling & Simulation **9**, 89–128 (2011)

13. M. Droske, W. Ring, A mumford-shah level-set approach for geometric image registration. SIAM journal on Applied Mathematics **66**, 2127–2148 (2006)

14. M. Droske, M. Rumpf, A variational approach to nonrigid morphological image registration. SIAM Journal on Applied Mathematics **64**, 668–687 (2004)

15. B. Fischer and J. Modersitzki, Fast diffusion registration, Contemp. Math., 313 (2002), pp. 117–129

16. B. Fischer, J. Modersitzki, Curvature based image registration. Journal of Mathematical Imaging and Vision **18**, 81–85 (2003)

17. B. Fischer, J. Modersitzki, Ill-posed medicine?an introduction to image registration. Inverse Probl. **24** (2008)

18. F. Gigengack, L. Ruthotto, M. Burger, C.H. Wolters, X. Jiang, K.P. Schafers, Motion correction in dual gated cardiac pet using mass-preserving image registration. IEEE transactions on medical imaging **31**, 698–712 (2012)

19. E. Haber, J. Modersitzki, Numerical methods for volume preserving image registration. Inverse problems **20**, 1621 (2004)

20. E. Haber, J. Modersitzki, Image registration with guaranteed displacement regularity. International Journal of Computer Vision **71**, 361–372 (2007)

21. S. Henn, A multigrid method for a fourth-order diffusion equation with application to image processing. SIAM Journal on Scientific Computing **27**, 831–849 (2005)

22. E. Hodneland, A. Lundervold, J. Rørvik, A.Z. Munthe-Kaas, Normalized gradient fields for nonlinear motion correction of dce-mri time series. Computerized Medical Imaging and Graphics **38**, 202–210 (2014)

23. W. Hu, Y. Xie, L. Li, and W. Zhang, A total variation based nonrigid image registration by combining parametric and non-parametric transformation models, Neurocomputing, 144 (2014), pp. 222–237

24. M. Ibrahim, K. Chen, C. Brito-Loeza, A novel variational model for image registration using gaussian curvature. Geometry, Imaging and Computing **1**, 417–446 (2014)

25. C.T. Kelley, *Iterative Methods for Optimization*, vol. 18 (Siam, 1999)

26. L. König, J. Rühaak, A fast and accurate parallel algorithm for non-linear image registration using normalized gradient fields, in Biomedical Imaging (ISBI), in 11th International Symposium on IEEE, vol. 2014 (IEEE, 2014), pp. 580–583

27. D. Loeckx, P. Slagmolen, F. Maes, D. Vandermeulen, P. Suetens, Nonrigid image registration using conditional mutual information. IEEE transactions on medical imaging **29**, 19–29 (2010)
28. L.M. Lui, T.C. Ng, A splitting method for diffeomorphism optimization problem using beltrami coefficients. Journal of Scientific Computing **63**, 573–611 (2015)
29. F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, P. Suetens, Multimodality image registration by maximization of mutual information. IEEE transactions on Medical Imaging **16**, 187–198 (1997)
30. A. Mang, G. Biros, An inexact newton-krylov algorithm for constrained diffeomorphic image registration. SIAM journal on imaging sciences **8**, 1030–1069 (2015)
31. A. Mang, G. Biros, Constrained h1-regularization schemes for diffeomorphic image registration. SIAM journal on imaging sciences **9**, 1154–1194 (2016)
32. J. Modersitzki, *Numerical Methods for Image Registration* (Oxford University Press on Demand, 2004)
33. J. Modersitzki, *FAIR: Flexible Algorithms for Image Registration*, vol. 6 (SIAM, 2009)
34. J. Nocedal, S.J. Wright, *Numerical Optimization 2nd* (2006)
35. F.P. Oliveira, J.M.R. Tavares, Medical image registration: a review. Computer methods in biomechanics and biomedical engineering **17**, 73–93 (2014)
36. C.C. Paige, M.A. Saunders, Solution of sparse indefinite systems of linear equations. SIAM journal on numerical analysis **12**, 617–629 (1975)
37. J.P. Pluim, J.A. Maintz, M.A. Viergever, Mutual-information-based registration of medical images: a survey. IEEE transactions on medical imaging **22**, 986–1004 (2003)
38. G. Roland, L.T. Patrick, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics* (SIAM, 1989)
39. J. Rühaak, L. König, M. Hallmann, N. Papenberg, S. Heldmann, H. Schumacher, B. Fischer, A fully parallel algorithm for multimodal image registration using normalized gradient fields, in Biomedical Imaging (ISBI), in *10th International Symposium on IEEE*, vol. 2013 (IEEE, 2013), pp. 572–575
40. A. Sotiras, C. Davatzikos, N. Paragios, Deformable medical image registration: A survey. IEEE transactions on medical imaging **32**, 1153–1190 (2013)
41. W. Sun, Y.-X. Yuan, *Optimization Theory and Methods: Nonlinear Programming*, vol. 1 (Springer Science & Business Media, 2006)
42. X.-C. Tai, J. Hahn, G.J. Chung, A fast algorithm for Euler's elastica model using augmented lagrangian method. SIAM Journal on Imaging Sciences **4**, 313–344 (2011)
43. A. Theljani, K. Chen, An augmented Lagrangian method for solving a new variational model based on gradients similarity measures and high order regulariation for multimodality registration. Inverse Problems and Imaging **13**, 309–335 (2019)
44. A. Theljani, K. Chen, A nash game based variational model for joint image intensity correction and registration to deal with varying illumination. Inverse Probl. **36** (2020)
45. A. Theljani, K. Chen, An unsupervised deep learning method for diffeomorphic mono-and multi-modal image registration, in *Medical Image Understanding and Analysis (MIUA 2019)*, vol. 1065 (Communications in Computer and Information Science, Springer, 2020)
46. P. Viola, W.M. Wells III, Alignment by maximization of mutual information. Int. J. Comput. Vis. **24**, 137–154 (1997)
47. C. Xing, P. Qiu, Intensity-based image registration by nonparametric local smoothing. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**, 2081–2092 (2011)
48. D. Zhang, K. Chen, A novel diffeomorphic model for image registration and its algorithm. Journal of Mathematical Imaging and Vision **60**, 1261–1283 (2018)
49. D. Zhang, A. Theljani, K. Chen, On a new diffeomorphic multi-modality image registration model and its convergent gauss-newton solver. Journal of Mathematical Research with Applications **39**, 633–656 (2019)
50. J. Zhang, K. Chen, Variational image registration by a total fractional-order variation model. Journal of Computational Physics **293**, 442–461 (2015)
51. J. Zhang, K. Chen, B. Yu, An improved discontinuity-preserving image registration model and its fast algorithm. Applied Mathematical Modelling **40**, 10740–10759 (2016)

52. J. Zhang, K. Chen, and B. Yu, A novel high-order functional based image registration model with inequality constraint, Computers & Mathematics with Applications, 72 (2016), pp. 2887–2899