INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 13, Number 6, Pages 969–985

A NEW CURVATURE-BASED IMAGE REGISTRATION MODEL AND ITS FAST ALGORITHM

JIN ZHANG AND KE CHEN

Abstract. Recently, Chumchob-Chen-Brito (2011) proposed the so-called mean curvature model which appeared to deliver better registration results for both smooth and non-smooth deformation fields than a large class of competing methods. However, the two displacement variables in a deformation field are regularized separately in their model and so coupling between them is not present. Therefore their mean curvature model has a weakness and may not yield the best registration results in some situations such as in non-smooth registration problems with non-axis-aligned discontinuities, as expected of a high order model. To design a new model based on interdependence between components of the deformation field, suitable for smooth and non-smooth registration problems, we propose a new vectorial curvature regularizer in this paper and present an iterative method for numerical solution of the resulting variational model. Experiments using both synthetic and realistic images confirm that the proposed model is more robust than the Chumchob-Chen-Brito (2011) model in registration quality for a wide range of examples.

Key words. Deformable registration, regularization, multilevel, curvature.

1. Introduction

In image processing, one is often interested not only in analyzing an image but also in comparing images in order to combine information or track changes. For this reason, image registration, also called image matching or image warping, is one of the most useful and challenging tasks in imaging applications. It is a problem frequently encountered in diverse application areas, such as astronomy, art, biology, chemistry, remote sensing and so on. Especially, in medical applications, noninvasive imaging is increasingly used in almost all stages of patient care: from disease detection to treatment guidance and monitoring. For comprehensive surveys of these applications, we refer to [8, 25], and references therein.

A general framework of image registration can be stated as follows: given two images of the same object, respectively the reference R (fixed) and the template T (moving), we search for a vector-valued transformation $\boldsymbol{\varphi}$ defined by

$$\boldsymbol{\varphi}(\boldsymbol{u})(\cdot) : \mathbb{R}^d \to \mathbb{R}^d,\ \boldsymbol{\varphi}(\boldsymbol{u})(\boldsymbol{x}) : \boldsymbol{x} \mapsto \boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})$$

or equivalently the unknown displacement field $\boldsymbol{u}$

$$\boldsymbol{u} : \mathbb{R}^d \to \mathbb{R}^d,\ \boldsymbol{u} : \boldsymbol{x} \mapsto \boldsymbol{u}(x) = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}), \cdots, u_d(\boldsymbol{x}))^\top,$$

such that the transformed template $T \circ \boldsymbol{\varphi}(\boldsymbol{u}(x)) = T(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) = T(\boldsymbol{u})$ becomes similar to the reference R. Here $d \in \mathbb{N}$ represents the spatial dimension of the images which is usually $d = 2$ in the two-dimensional case or $d = 3$ the three-dimensional case with boundary $\partial\Omega$. Without loss of generality, we focus on $d = 2$ and state that registration results are readily extendable to $d = 3$. Then $\boldsymbol{x} = (x_1, x_2)$ and $d\Omega = dx_1 dx_2$. When the corresponding location $\boldsymbol{\varphi}(\boldsymbol{u}(\boldsymbol{x})) = \boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})$ is calculated for each spatial location $\boldsymbol{x}$ in the image domain $\Omega \subset \mathbb{R}^2$, an image interpolation is required to assign the image intensity values for the transformed template $T(\boldsymbol{u})$

Received by the editors January 1, 2015 and, in revised form, March 22, 2016.
2000 *Mathematics Subject Classification.* 65F10, 65M55, 68U10.
This work was supported by the United Kingdom EPSRC grant (number EP/K036939/1) .

at non-grid locations within image boundaries. For location outside the image boundaries, the image intensities are usually set to be a constant value, typically zero, we can refer to [25]. It is well worth noticing that the displacement $\boldsymbol{u}$ is more intuitive than the transformation $\boldsymbol{\varphi}$ because it can measure how much a point in the transformed template $T(\boldsymbol{u})$ has moved away from its original position in T. Here we shall restrict ourselves to scalar or gray intensity images and model them as compactly supported functions mapping from the image domain $\Omega \subset \mathbb{R}^2$ into $V \subset \mathbb{R}_0^+$. Assuming the image intensities of R and T are comparable, the image registration problem can be formulated as the following similarity minimization

$$\min_{\mathbf{u}} \Big\{ \mathcal{D}(\boldsymbol{u}) = \frac{1}{2} \int_\Omega \left(T(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) - R(\boldsymbol{x})\right)^2 d\Omega \Big\}. \tag{1}$$

It has long been known that image registration (or the above model) is an ill-conditioned problem [25]. As a consequence, regularization is inevitable. A common treatment of the registration problem is based on the Tikhonov type regularization approach: find the transformation $\boldsymbol{u}$ by minimizing the joint energy functional

$$\mathcal{J}_\alpha(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha \mathcal{R}(\boldsymbol{u}) \tag{2}$$

where $\mathcal{R}(\boldsymbol{u})$ is a deformation regularizer and $\alpha > 0$ is the regularization parameter that compromises similarity and regularity.

Non-surprisingly, the choice of $\mathcal{R}(\boldsymbol{u})$ is very crucial for effective registration. For instance, $\mathcal{R}(\boldsymbol{u}) = \|\boldsymbol{u}\|_2$ is sufficient to ensure (2) to be well defined but it does not lead to a useful model. There exist many models for deformable image registration, mainly differing in how regularization is introduced. For smooth registration problems (where $\boldsymbol{u}$ can be assumed to be smooth), the common regularization techniques such as diffusion-, elastic-, or curvature-based image registration are known to generate globally smooth deformation fields, see more details in [11, 13, 14, 24, 23, 22, 25, 30] and references therein. However, these techniques become poor if discontinuities or steep gradients in the deformation fields are required. The total variation-based (TV) image registration is better for preserving discontinuities of the deformation fields [15, 16, 28]. Nevertheless, the TV model may not be appropriate for smooth registration problems. Clearly $\boldsymbol{u}$ is unknown and any assumption on it is practically unrealistic; therefore a common wish is for a model to achieve robustness regardless of what properties $\boldsymbol{u}$ may have.

Recently, Chumchob-Chen-Brito [12] proposed the so-called mean curvature-based variational model

$$\min_{\boldsymbol{u}} \mathcal{J}_\alpha(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha \mathcal{R}^{\mathrm{CCBc}}(\boldsymbol{u}), \qquad \mathcal{R}^{\mathrm{CCBc}}(\boldsymbol{u}) = \sum_{l=1}^{2} \int_\Omega \phi(\kappa_{\mathrm{CCB}}(u_l)) d\Omega. \tag{3}$$

Here $\phi(s) = \frac{1}{2}s^2$ and $\kappa_{\mathrm{CCB}}(u_l) = \nabla \cdot \frac{\nabla u_l}{|\nabla u_l|_\beta}$ with $|\nabla u_l|_\beta = \sqrt{((u_l)_x)^2 + ((u_l)_y)^2 + \beta}$, and $\beta > 0$ is a small real parameter for avoiding non-differentiability where $|\nabla u_l| = 0$ [12, 15, 16, 28]. The work of [12] showed that $\mathcal{R}^{\mathrm{CCBc}}$ shares some attractive properties with the Fischer-Modersitzki's linear curvature-based regularizer [14, 24, 25], e.g. the energy functional (3) does not penalize affine-linear transformations and it is invariant under planar rotation and translation. Moreover, model (3) appeared to deliver excellent results for both smooth and non-smooth registration problems, provided that u_1 and u_2 are not strongly coupled. This is because the nonlinear diffusion processes resulting from the first variation of $\mathcal{R}^{\mathrm{CCBc}}(\boldsymbol{u})$ do not enforce coupling between the primary components of the deformation field, u_1 and u_2. Thus their mean curvature model may not obtain a good registration in some

situations, such as non-smooth registration problems with non-axis-aligned discontinuities. Motivated by several regularization techniques that have been proved to be very useful in vector-valued image denoising [4, 5, 7] and in optical flow computation [2, 3], we propose a new vectorial curvature model for image registration in this paper.

Although related regularization techniques have been used for color image denoising, to the best of our knowledge, they have not been studied thoroughly yet for the registration problem (2), especially for solving smooth and non-smooth registration problems. We summaries the advantages of the new model as follows: 1) these vectorial regularization methods lead to desirable properties, e.g. they preserve discontinuities in the deformation field for non-smooth registration problems and smoothness for smooth registration problems; 2) they incorporate the coupling information between two primary components of the deformation field so that no priori assumption of deformation is needed.

In general, the optimization problem (2) cannot be solved analytically, thus numerical schemes and appropriate discretizations are required. Developing an efficient numerical solution of the registration problem is an important task. Over the past decades, there are two commonly used types of numerical schemes to compute a numerical solution of the minimization problem (2) for a given regularization parameter α. The first approach, the optimize-discretize approach, first forms the objective function and then derives the continuous Euler-Lagrange equations, which are finally discretized and solved numerically; see, e.g., [12, 25, 11, 13, 30, 14, 24, 28]. The second approach is the discretize-optimize approach which aims to discretize the joint functional $\mathcal{J}_\alpha$ in (2) and then solve the discrete minimization problem by standard optimization methods; see, e.g. [22, 23, 21, 20, 19]. In this paper, we take the latter discretize-optimize approach. Thus our framework is related to some previous works of [22, 23] where an elastic (first order) rather than a higher-order regularizer was used. We remark that, though a small smoothing parameter β is desirable in (3), smaller β corresponds to stronger nonlinearity and slows down the convergence of many numerical methods. Chumchob-Chen-Brito [12] developed a convergent multigrid method using a local primal-dual fixed-point method as as a smoother to solve (2), provided that the smoothing parameter β is large enough (e.g. $\beta \geq 10^{-2}$). In contrast, our new model can be solved for very small smoothing parameter β (e.g. $\beta \leq 10^{-16}$).

The rest of the paper is organized as follows. In Section 2, we present a new curvature model suitable for both smooth and non-smooth deformation problems. Then in Section 3, we develop a numerical method based on fixed-point methods, a Gauss-Newton scheme with Armijo's Line Search and a multilevel method to achieve fast convergence. Experimental results both syntectic and real images are illustrated in Section 4 before conclusions are made in Section 5.

2. A new curvature regularizer based image registration model

Our ideal regulariser is expected to be related to the mean curvatures $\kappa(u_1), \kappa(u_2)$ as used by [12], but yet allows coupling between two primary components u_1, u_2 of the deformation field to further improve the registration quality for both smooth and non-smooth registration problems. One possible method is to introduce the model

$$\min_{\boldsymbol{u}} \mathcal{J}_\alpha(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha\mathcal{R}(\boldsymbol{u}), \qquad \mathcal{R}(\boldsymbol{u}) = \int_\Omega \sqrt{\kappa(u_1)^2 + \kappa(u_2)^2} d\Omega$$

which, unfortunately, is not the best approach due to enforcing extremely strong coupling. Other possibilities may be based on the work of [7]. Below we make use of the modified mean curvatures $\kappa_1(\boldsymbol{u}), \kappa_2(\boldsymbol{u})$. Our motivations are based on several regularization techniques that have been proved to be very useful in vector-valued image denoising [4, 5, 7] and in optical flow computation [2, 3].

To this end, we first note that minimizing the TV semi-norm

$$\int_\Omega |\nabla u| d\Omega$$

for a scalar function u leads to the mean curvature [28]

$$\kappa(u) = \kappa_{CCB}(u) = \nabla \cdot \frac{\nabla u}{|\nabla u|}.$$

We next note that minimizing the vectorial TV semi-norm [4]

$$\int_\Omega |\nabla \boldsymbol{u}| d\Omega \equiv \int_\Omega \sqrt{|\nabla u_1|^2 + |\nabla u_2|^2} d\Omega$$

leads to two new mean curvature like quantities

$$\kappa_1(\boldsymbol{u}) = \nabla \cdot \frac{\nabla u_1}{|\nabla \boldsymbol{u}|},\ \kappa_2(\boldsymbol{u}) = \nabla \cdot \frac{\nabla u_2}{|\nabla \boldsymbol{u}|} \tag{4}$$

which clearly have built-in coupling between u_1, u_2 through $|\nabla \boldsymbol{u}|$.

Therefore, to make full use of the coupling information, we propose a new variational model based on (4)

$$\min_{\boldsymbol{u}} \mathcal{J}_\alpha(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha \mathcal{R}^{\mathrm{NewC}}(\boldsymbol{u}), \qquad \mathcal{R}^{\mathrm{NewC}}(\boldsymbol{u}) = \frac{1}{2} \sum_{l=1}^{2} \int_\Omega \left(\nabla \cdot \frac{\nabla u_l}{|\nabla \boldsymbol{u}|_\beta}\right)^2 d\Omega \tag{5}$$

where $|\nabla \boldsymbol{u}|_\beta = \sqrt{|\nabla u_1|^2 + |\nabla u_2|^2 + \beta}$ with β a small positive parameter for avoiding singularities when $|\nabla \boldsymbol{u}| = 0$. That is, our vectorial curvature model takes the form

$$\min_{\boldsymbol{u}} \Big\{ \mathcal{J}_\alpha(\boldsymbol{u}) = \frac{1}{2} \int_\Omega \big(T(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) - R(\boldsymbol{x})\big)^2 d\Omega + \frac{\alpha}{2} \sum_{l=1}^{2} \int_\Omega \left(\nabla \cdot \frac{\nabla u_l}{|\nabla \boldsymbol{u}|_\beta} \right)^2 d\Omega \Big\}. \tag{6}$$

Our particular choice of regularizer in (6) has advantages. Firstly, the new curvature regularization $\mathcal{R}^{\mathrm{NewC}}$ shares some attractive properties with Fischer-Modersitzki's curvature-based regularization and $\mathcal{R}^{\mathrm{CCBc}}$ (e.g. affine transform and invariance). Secondly, as intended, a visually pleasing registration result can be obtained by using $\mathcal{R}^{\mathrm{NewC}}$ for non-smooth registration problems with non-axis-aligned discontinuities. Finally, due to the above two advantages, it is worth stating that it is expected to outperform $\mathcal{R}^{\mathrm{CCBc}}$ which was previously the best.

We remark that the Euler Lagrange equations for (6) will be of fourth order and highly nonlinear and subsequent numerical solutions would be a numerically challenging task. If one takes the optimize-discretize approach, the ideas from [6, 7, 12] have to be used to develop convergent numerical methods. Below we consider the alternative discretize-optimize approach.

3. Numerical solution of the image registration model (6)

In general, an optimization problem such as (6) cannot be solved analytically; thus numerical schemes and appropriate discretizations are required. In the discretize-optimize method, in order to take advantage of efficient optimization techniques, all parts of the discrete problem need to be continuously differentiable which is

true if β is not zero (though small). In this section, we first discuss briefly the discretization we use, and then describe the details of numerical algorithms.

3.1. Finite difference discretization. We assume that our discrete images have $m_1 \times m_2$ pixels. For the sake of simplicity, we further assume that the image domain $\Omega = [0, \omega_1] \times [0, \omega_2] \subset \mathbb{R}^2$, then each side of each pixel has length $h_i = \omega_i / m_i, i = 1, 2$. Let the discrete domain in Ω be denoted by

$$\Omega_h = \{\boldsymbol{x} \Big| \boldsymbol{x} = \begin{pmatrix} x_{1_{j_1}} \\ x_{2_{j_2}} \end{pmatrix} = \begin{pmatrix} (j_1 - 0.5)h_1 \\ (j_2 - 0.5)h_2 \end{pmatrix},\ j_1 = 1, 2, \cdots, m_1;\ j_2 = 1, 2, \cdots, m_2\}.$$

Points in this set can be arranged into vector form. Let

$$\begin{aligned} \boldsymbol{x}_{c,1} =& [x_{1_{1,1}}, x_{1_{2,1}}, \cdots, x_{1_{m_1,1}}, x_{1_{1,2}}, x_{1_{2,2}}, \cdots, \\ & x_{1_{m_1,2}}, \cdots, x_{1_{1,m_2}}, x_{1_{2,m_2}}, \cdots, x_{1_{m_1,m_2}}]^\top, \\ \boldsymbol{x}_{c,2} =& [x_{2_{1,1}}, x_{2_{2,1}}, \cdots, x_{2_{m^1,1}}, x_{2_{1,2}}, x_{2_{2,2}}, \cdots, \\ & x_{2_{m_1,2}}, \cdots, x_{2_{1,m_2}}, x_{2_{2,m_2}}, \cdots, x_{2_{m_1,m_2}}]^\top, \end{aligned}$$

and $\boldsymbol{X}_c^h = [\boldsymbol{x}_{c,1}; \boldsymbol{x}_{c,2}]$.

3.1.1. Discretization of the new regularizer $\mathcal{R}^{\mathbf{NewC}}(\boldsymbol{u})$. We denote the discrete analogue of the continuous displacement field $\boldsymbol{u} = (u_1, u_2)^\top$ in the discrete domain Ω_h by $\boldsymbol{u}^h = (u_1^h, u_2^h)^\top$, where u_1^h and u_2^h are grid functions: $(u_l^h)_{j_1,j_2} = u_l^h(x_{1_{j_1}}, x_{2_{j_2}})$, $j_1 = 1, 2, \cdots, m_1;\ j_2 = 1, 2, \cdots, m_2$ and $l = 1, 2$. Since the new curvature regularizers $\mathcal{R}^{\mathrm{NewC}}(\boldsymbol{u})$ is expressed in terms of the more complex differential operators gradient ∇ and divergence $\nabla\cdot$, we introduce the notation ∇^h and $\nabla^h\cdot$ for their discrete analogues. The discrete gradient operator ∇^h at each pixel (j_1, j_2) is defined by

$$(\nabla^h \boldsymbol{u}^h)_{j_1,j_2} = ((\nabla^{h,1} u_1^h)_{j_1,j_2}, (\nabla^{h,2} u_2^h)_{j_1,j_2})^\top$$

with

$$\begin{aligned} (\nabla^{h,l} u_l^h)_{j_1,j_2} &= ((\partial_1^{h,l} u_l^h)_{j_1,j_2}, (\partial_2^{h,l} u_l^h)_{j_1,j_2})^\top \\ (\partial_1^{h,l} u_l^h)_{j_1 j_2} &= \begin{cases} (u_l^h)_{j_1+1,j_2} - (u_l^h)_{j_1,j_2}, & \text{if } j_1 < m_1 \\ 0, & \text{if } j_1 = m_1, \end{cases} \\ (\partial_2^{h,l} u_l^h)_{j_1 j_2} &= \begin{cases} (u_l^h)_{j_1,j_2+1} - (u_l^h)_{j_1,j_2}, & \text{if } j_2 < m_2 \\ 0, & \text{if } j_2 = m_2. \end{cases} \end{aligned}$$

Here homogeneous Neumann boundary conditions on $\boldsymbol{u}$ are assumed:

$$\frac{\partial u_l}{\partial \boldsymbol{\nu}} = 0,\ l = 1, 2 \text{ on } \partial\Omega.$$

The discrete divergence operator is the negative adjoint of the gradient operator due to the analysis of the continuous setting, namely $\nabla\cdot = -\nabla^*$. Therefore, it can be defined by the following:

$$\begin{aligned} (\nabla \cdot w_l)_{j_1,j_2} =& \begin{cases} (w_l^1)_{j_1,j_2} - (w_l^1)_{j_1-1,j_2} \\ (w_l^1)_{j_1,j_2} \\ -(w_l^1)_{j_1-1,j_2} \end{cases} \\ &+ \begin{cases} (w_l^2)_{j_1,j_2} - (w_l^2)_{j_1,j_2-1} & \text{if } 1 < j_i < m_i\ , i = 1, 2, \\ (w_l^2)_{j_1,j_2} & \text{if } j_1 = j_2 = 1, \\ -(w_l^2)_{j_1,j_2-1} & \text{if } j_1 = m_1\ ,\ j_2 = m_2. \end{cases} \end{aligned}$$

For later convenience, we re-arrange the grid functions u_1^h and u_2^h as column vectors $\boldsymbol{u}_1^h$ and $\boldsymbol{u}_2^h$ according to lexicographical ordering, respectively

$$\boldsymbol{u}_1^h = (u^h_{1_{1,1}}, u^h_{1_{2,1}}, \cdots, u^h_{1_{m_1,1}}, u^h_{1_{1,2}}, u^h_{1_{2,2}}, \cdots, u^h_{1_{m_1,2}}, \cdots, u^h_{1_{1,m_2}}, u^h_{1_{2,m_2}}, \cdots, u^h_{1_{m_1,m_2}})^\top,$$
$$\boldsymbol{u}_2^h = (u^h_{2_{1,1}}, u^h_{2_{2,1}}, \cdots, u^h_{2_{m_1,1}}, u^h_{2_{1,2}}, u^h_{2_{2,2}}, \cdots, u^h_{2_{m_1,2}}, \cdots, u^h_{2_{1,m_2}}, u^h_{2_{2,m_2}}, \cdots, u^h_{2_{m_1,m_2}})^\top.$$

Then $\boldsymbol{u}_1^h \in \mathbb{R}^N$, $\boldsymbol{u}_2^h \in \mathbb{R}^N$ and $\boldsymbol{U}^h = (\boldsymbol{u}_1^h; \boldsymbol{u}_2^h) \in \mathbb{R}^{2N}$, where $N = m_1 m_2$. The discrete gradient $(\nabla^{h,l} u_l^h)_{j_1,j_2}$ can be expressed by a multiplication of the matrix $A_k^\top \in \mathbb{R}^{2\times N} (k = 1, 2, \cdots, N)$ to the vector $\boldsymbol{u}_l^h (l = 1, 2)$:

$$A_k^\top \boldsymbol{u}_l^h = \begin{cases} ((\boldsymbol{u}_l^h)_{k+1} - (\boldsymbol{u}_l^h)_k; (\boldsymbol{u}_l^h)_{k+m_2} - (\boldsymbol{u}_l^h)_k), \text{ if } k \bmod m_1 \neq 0 \text{ and } k + m_2 \leq N \\ (0; (\boldsymbol{u}_l^h)_{k+m_2} - (\boldsymbol{u}_l^h)_k), \text{ if } k \bmod m_1 = 0 \text{ and } k + m_2 \leq N \\ ((\boldsymbol{u}_l^h)_{k+1} - (\boldsymbol{u}_l^h)_k; 0), \text{ if } k \bmod m_1 \neq 0 \text{ and } k + m_2 > N \\ (0; 0), \text{ if } k \bmod m_1 = 0 \text{ and } k + m_2 > N. \end{cases}$$

By concatenating the matrices A_k, $k = 1, 2, \cdots, N$, we define

$$A = (A_1, A_2, \cdots, A_N) = (A_{1,1}, A_{1,2}, \cdots, A_{N,1}, A_{N,2}) \in \mathbb{R}^{N\times 2N};$$
$$A_x = (A_{1,1}, A_{2,1}, \cdots, A_{N,1}) \in \mathbb{R}^{N\times N}; \qquad A_y = (A_{1,2}, A_{2,2}, \cdots, A_{N,2}) \in \mathbb{R}^{N\times N};$$
$$\nabla^{h,1}\boldsymbol{u}_1^h = \begin{bmatrix} A_x{}^\top \\ A_y{}^\top \end{bmatrix} \boldsymbol{u}_1^h \triangleq B\boldsymbol{u}_1^h; \qquad \nabla^{h,2}\boldsymbol{u}_2^h = \begin{bmatrix} A_x{}^\top \\ A_y{}^\top \end{bmatrix} \boldsymbol{u}_2^h \triangleq B\boldsymbol{u}_2^h.$$

Thus, for the discrete gradient operator ∇^h, we have

$$\nabla^h \boldsymbol{U}^h = \begin{bmatrix} \nabla^{h,1} & 0 \\ 0 & \nabla^{h,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1^h \\ \boldsymbol{u}_2^h \end{bmatrix} = \begin{bmatrix} B & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1^h \\ \boldsymbol{u}_2^h \end{bmatrix} \triangleq \boldsymbol{A}\boldsymbol{U}^h.$$

The discrete divergence operator is the negative adjoint of the gradient operator due to the analysis of the continuous setting, namely $\nabla\cdot = -\nabla^*$.

Now we consider the main operator

$$\mathcal{R}^{\mathrm{NewC}}(\boldsymbol{u}) = \frac{1}{2}\int_\Omega \mathcal{B}[\boldsymbol{u}] d\Omega, \qquad \mathcal{B}[\boldsymbol{u}] = (\nabla \cdot \frac{\nabla u_1}{|\nabla \boldsymbol{u}|_\beta})^2 + (\nabla \cdot \frac{\nabla u_2}{|\nabla \boldsymbol{u}|_\beta})^2. \tag{7}$$

The discrete form of the above is the following

$$\mathcal{R}^{h\mathrm{NewC}}(\boldsymbol{U}^h) = \frac{1}{2} h_d \sum_{i,j} (\mathcal{B}[\boldsymbol{u}^h])_{i,j} = \frac{1}{2} h_d \mathbb{B}^h[\boldsymbol{U}^h] \tag{8}$$

where $h_d = h_1 h_2$ and

$$\begin{aligned} \mathbb{B}^h[\boldsymbol{U}^h] &= \Big|\frac{-B^\top B\boldsymbol{u}_1^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta}\Big|^2 + \Big|\frac{-B^\top B\boldsymbol{u}_2^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta}\Big|^2 \\ &= \frac{(\boldsymbol{u}_1^h)^\top B^\top BB^\top B\boldsymbol{u}_1^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2} + \frac{(\boldsymbol{u}_2^h)^\top B^\top BB^\top B\boldsymbol{u}_2^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2} \\ &= \frac{1}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}((\boldsymbol{u}_1^h)^\top B^\top BB^\top B\boldsymbol{u}_1^h + (\boldsymbol{u}_2^h)^\top B^\top BB^\top B\boldsymbol{u}_2^h) \\ &= \frac{1}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}\Big((\boldsymbol{u}_1^h)^\top, (\boldsymbol{u}_2^h)^\top\Big) \begin{bmatrix} B^\top BB^\top B & 0 \\ 0 & B^\top BB^\top B \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1^h \\ \boldsymbol{u}_2^h \end{bmatrix} \\ &= \frac{1}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}(\boldsymbol{U}^h)^\top \begin{bmatrix} B & 0 \\ 0 & B \end{bmatrix}^\top \begin{bmatrix} B & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} B & 0 \\ 0 & B \end{bmatrix}^\top \begin{bmatrix} B & 0 \\ 0 & B \end{bmatrix} \boldsymbol{U}^h \\ &= \frac{(\boldsymbol{U}^h)^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{U}^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}. \end{aligned}$$

Further

$$\mathcal{R}^{h\mathrm{NewC}}(\boldsymbol{U}^h) = \frac{h_d}{2}\frac{(\boldsymbol{U}^h)^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{U}^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}. \tag{9}$$

3.1.2. Discretization of images T and R. Linear interpolation which can be evaluated with low computational costs is a reasonable tool in image registration. However, although the interpolation is differentiable almost everywhere, it is not differentiable at the grid points. In order to benefit from fast and efficient optimization schemes, smoother interpolation are needed. For noisy image we use a smoothing B-spline to approximation the images where the smoothing parameter can be chosen accordingly if the noise level of the data is known a priori; see [18, 33]. In our implementation we use a cubic B-spline approximation. The continuous smooth approximations for template T and reference R are denoted by $\boldsymbol{\mathcal{T}}$ and $\boldsymbol{\mathcal{R}}$, respectively. For further discussion on the effects of higher or lower order B-spline interpolation on the quality of the registration, we can refer to [31].

We can get the discrete reference image

$$\vec{R} = \boldsymbol{\mathcal{R}}(\boldsymbol{X}_c^h) \tag{10}$$

and the discrete transformed template image

$$\vec{T}(\boldsymbol{U}^h) = \boldsymbol{\mathcal{T}}(\boldsymbol{X}_c^h + \boldsymbol{U}^h). \tag{11}$$

Note that $\vec{T}(\boldsymbol{U}^h)$ is the discrete analogue of the transformed template image $T(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x}))$ as a function of $\boldsymbol{u}$. We denote the Jacobian of $\vec{T}$ by

$$\vec{T}_{\boldsymbol{U}^h} = \frac{\partial \vec{T}}{\partial U^h}(U^h) = \frac{\partial \boldsymbol{\mathcal{T}}}{\partial \boldsymbol{U}_c^h}(\boldsymbol{U}_c^h)$$

where $\boldsymbol{U}_c^h = \boldsymbol{X}_c^h + \boldsymbol{U}^h$, and the Jacobian of $\boldsymbol{\mathcal{T}}$ is a block matrix with diagonal blocks.

3.1.3. Discretization of the similarity term $\mathcal{D}$. In the discrete analogue, the integral is approximated by a midpoint quadrature. According to (10) and (11) our discretization of the similarity (distance) measure $\mathcal{D}$ in (1) is straightforward:

$$\mathcal{D}^h(\boldsymbol{U}^h) = \frac{h_d}{2}(\vec{T}(\boldsymbol{U}^h) - \vec{R})^\top(\vec{T}(\boldsymbol{U}^h) - \vec{R})$$

and the derivative of the discretized functional $\mathcal{D}^h(\boldsymbol{U}^h)$ with respect to $\boldsymbol{U}^h$ is computed by

$$d\mathcal{D}^h(\boldsymbol{U}^h) = h_d(\vec{T}(\boldsymbol{U}^h) - \vec{R})^\top \vec{T}_{\boldsymbol{U}^h}$$

In addition, the second derivative $d^2\mathcal{D}^h(\boldsymbol{U}^h)$ of $\mathcal{D}$ can also be calculated straightforwardly

$$d^2\mathcal{D}^h(\boldsymbol{U}^h) = h_d(\vec{T}_{\boldsymbol{U}^h})^\top \vec{T}_{\boldsymbol{U}^h}. \tag{12}$$

3.2. Solution of the discrete optimization problem. The discretized analogue of the image registration problem (6) reads as follows:

$$\min_{\boldsymbol{U}^h}\{\mathcal{J}_\alpha(\boldsymbol{U}^h) = \mathcal{D}^h(\boldsymbol{U}^h) + \alpha\mathcal{R}^{h\mathrm{NewC}}(\boldsymbol{U}^h)\} \tag{13}$$

i.e.

$$\min_{\boldsymbol{U}^h}\{\mathcal{J}_\alpha(\boldsymbol{U}^h) = \mathcal{D}^h(\boldsymbol{U}^h) + \frac{\alpha}{2}h_d\frac{(\boldsymbol{U}^h)^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{U}^h}{|\boldsymbol{A}\boldsymbol{U}^h|_\beta^2}.$$

Obviously, the above functional in an algebraic form is nonlinear. In subsequent solutions, we need to differentiate it twice. To facilitate differentiation, we shall introduce a lagging into the denominator of the new curvature regularizer $\mathcal{R}^{h\mathrm{NewC}}(\boldsymbol{U}^h)$. The lagged quantity in (13) uses a previous and known iterate $\boldsymbol{U}^{h^{(k)}}$. We note that the lagging method by 'frozen coefficients' is well known for variational approaches, e. g. see [32, 29, 10, 9] when related to the total variation (TV) operator. Thus we obtain the following form

$$\min_{\boldsymbol{U}^h}\Big\{\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^h)=\mathcal{D}^h(\boldsymbol{U}^h)+\frac{\alpha}{2}h_d\frac{(\boldsymbol{U}^h)^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{U}^h}{|\boldsymbol{A}\boldsymbol{U}^{h^{(k)}}|_\beta^2}\Big\}. \tag{14}$$

This is equivalent to replacing $\mathcal{J}_\alpha$ by a quasi-quadratic $\hat{\mathcal{J}}_\alpha$ obtained in a Taylor expansion,

$$\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k)}}+\delta_{\boldsymbol{U}^h})\approx\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}}+\delta_{\boldsymbol{U}^h})=\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}})+d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}})\delta_{\boldsymbol{U}^h}+\frac{1}{2}\delta_{\boldsymbol{U}^h}^\top\boldsymbol{H}\delta_{\boldsymbol{U}^h}$$

where the gradient $d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^h)$ at the known iterate $\boldsymbol{U}^{h^{(k)}}$ is

$$d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}})=d\mathcal{D}^h(\boldsymbol{U}^{h^{(k)}})+\alpha h_d\frac{(\boldsymbol{U}^{h^{(k)}})^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{A}^\top\boldsymbol{A}}{|\boldsymbol{A}\boldsymbol{U}^{h^{(k)}}|_\beta^2} \tag{15}$$

and $\boldsymbol{H}$, positive and semi-definite, is an approximation to the Hessian matrix

$$\boldsymbol{H}=d^2\mathcal{D}^h(\boldsymbol{U}^{h^{(k)}})+\alpha h_d\frac{\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{A}^\top\boldsymbol{A}}{|\boldsymbol{A}\boldsymbol{U}^{h^{(k)}}|_\beta^2}. \tag{16}$$

The approximation is due to lagging in the regularization term. Note $\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}})=\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k)}})$.

Hence for (14) , the Gauss-Newton scheme takes the form

$$\boldsymbol{U}^{h^{(k+1)}}=\boldsymbol{U}^{h^{(k)}}+\delta_{\boldsymbol{U}^h},\qquad\boldsymbol{H}\delta_{\boldsymbol{U}^h}=-d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}}). \tag{17}$$

In order to ensure a sufficient decrease in the objective function $\mathcal{J}_\alpha(\boldsymbol{U}^h)$, a line search procedure is required; see [27]. So equation (17) becomes

$$\boldsymbol{U}^{h^{(k+1)}}=\boldsymbol{U}^{h^{(k)}}+t\delta_{\boldsymbol{U}^h},\quad\boldsymbol{H}\delta_{\boldsymbol{U}^h}=-d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}}). \tag{18}$$

The numerical scheme is summarized in Algorithm 1.

In this paper, we use the standard Armijo line search. This procedure is iterated until the stopping criteria are satisfied. Here the Armijo Line Search can be briefly explained as follows. Starting with $t=1$, the objective function $\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k+1)}})$ of the new iterate $\boldsymbol{U}^{h^{(k+1)}}=\boldsymbol{U}^{h^{(k)}}+t\delta_{\boldsymbol{U}^h}$ is compared to the old value $\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k)}})$. If the reduction is not sufficient, the procedure is iterated, replacing t by $\frac{1}{2}t$. The standard criterion

$$\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k+1)}})<\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k)}})+\ t\ ((d\hat{\mathcal{J}}_\alpha(\boldsymbol{U}^{h^{(k)}}))^\top\boldsymbol{U}^{h^{(k)}})\ tol$$

where the standard choice $tol=10^{-4}$ is used. As a safeguard, the line search as well as the iteration is terminated if the step length t becomes too small. In this situation, the optimization algorithm is declared to have failed to converge. The algorithm is summarized in Algorithm 2. Denote $\mathcal{J}_c=\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k+1)}})$ for the accepted t, $\mathcal{J}_{init}=\mathcal{J}_\alpha(\boldsymbol{0}^h)$ and $\mathcal{J}_{old}=\mathcal{J}_\alpha(\boldsymbol{U}^{h^{(k)}})$. The following common stopping criteria for optimization are used to ensure a robust convergence; see [17].

Algorithm 1: Gauss-Newton for image registration: $\boldsymbol{u} \leftarrow \text{GNIRArmijo}(\alpha, \boldsymbol{u})$

```
Set iter = 0;
while true do
    Compute J_α(u), dĴ_α(u) and H using (14), (15) and (16), respectively;
    Update the iteration count: iter←iter + 1;
    Check the stopping rules;
    Solve the quasi-Newton's equation: Hδ_u = −dĴ_α(u) by using a
    preconditioned conjugate gradient method;
    Perform an Armijo Line Search: u_t ← Armijo(α, δ_u, u) ;
    if line search fail then
        break;
    else
        Update current values: u ← u_t;
    end
end
```

Algorithm 2: Armijo Line Search: $\boldsymbol{u} \leftarrow \text{Armijo}(\alpha, \delta_{\boldsymbol{u}}, \boldsymbol{u})$

```
Compute J_α(u) and dĴ_α(u) using (14) and (15), respectively;
Set k ← 0, t ← 1, MaxIter ← 10, and η ← 10^{-4};
while true do
    Set u_t ← u + tδ_u;
    Compute J_α(u_t) using (14);
    If J_α(u_t) < J_α(u) + tη(dĴ_α(u))^⊤ δ_u or k > MaxIter, break, end;
    Set t ← t/2 and k ← k + 1;
end
Set u ← u_t.
```

(1) Stop(1): $|\mathcal{J}_{\text{old}} - \mathcal{J}_{\text{c}}| \leq 10^{-3}(1 + |\mathcal{J}_{\text{init}}|)$ and Stop(2): $\|\boldsymbol{u}_{\text{c}} - \boldsymbol{u}_{\text{old}}\| \leq 10^{-2}(1 + \boldsymbol{u}0)$ and
Stop(3): $\|d\mathcal{J}_c\| \leq 10^{-2}(1 + |\mathcal{J}_{\text{init}}|)$
(2) or Stop(4): $\|d\hat{\mathcal{J}}_c\| \leq \epsilon$ or Stop(5): (iter $\geq$ maxIter).

Here $\boldsymbol{u}0$ is the initial guess on level k. The first criterion measures the relative variation in the objective function, the second criterion measures the relative variation in the parameters, and the third measures the norm of the gradient. If all these quantities are small, the iteration is terminated. The last two criteria $\|d\hat{\mathcal{J}}_c\| \leq \epsilon$ and iter$\geq$maxIter are quite natural, where $\epsilon = 10^{-16}$ denotes the machine precision and maxIter is an a priori chosen number. See, e.g., [17, 26] for a detailed discussion. Here maxIter is chosen to be small on finer levels for efficiency consideration and the largest on the coarsest level.

In order to save computational work and to speed up convergence, we combine the Gauss-Newton method with the multilevel scheme to solve (14). Starting on a coarse level where computations are cheap, a starting guess for a finer level is computed. On the fine level, only a very few correction steps are expected. Using a multilevel approach offers two major advantages. First, the optimization problems are easier to solve on the coarser levels, i.e., less iterations are needed to compute a minimizer. Second, details are diminished on coarser levels and the optimization

is thus more robust, and the risk of being trapped in local minima is reduced. In order to take advantage of efficient optimization techniques, all parts of the discrete problem need to be continuously differentiable. Thus multilevel representation of given images is necessary. The objective of multilevel representation is to derive a family of continuous models for given images. In order to easily understand, we give a simple example. Suppose that $m_1 = m_2 = 2^L$ with $L \in \mathbb{N}$ and $\mathrm{T} \in \mathbb{R}^{m_1 \times m_2}$ is given. A multilevel representation of the data T is $\{\mathrm{T}^l, l = 1, 2, \cdots, L\}$, where $\mathrm{T}^{\mathrm{L}} = \mathrm{T}$ and for $l = L : -1 : 1$,

$$\begin{aligned} \mathrm{T}^{l-1}(1 : 2^{l-1}, :) &= (\mathrm{T}^l(1 : 2 : \mathrm{end}, :) + \mathrm{T}^l(2 : 2 : \mathrm{end}, :))/2; \\ \mathrm{T}^{l-1}(:, 1 : 2^{l-1}) &= (\mathrm{T}^l(:, 1 : 2 : \mathrm{end}) + \mathrm{T}^l(:, 2 : 2 : \mathrm{end}))/2 \end{aligned} \tag{19}$$

We next summarize the multilevel scheme in Algorithm 3. In this Algorithm, bilinear interpolation techniques are used for the interpolation operator denoted by I_H^h.

Algorithm 3: Multilevel Image Registration: $\boldsymbol{u} \leftarrow$ MLIR(MLData)

Maxlevel$\leftarrow L$, % The finest level;
Minlevel$\leftarrow 3$, % The coarsest level;
Input MLData, % Multilevel representation of given images R and T according to (19);
for $l =$ Minlevel:Maxlevel **do**
  if $l ==$ Minlevel **then**
    Providing the first guess $\boldsymbol{u}_0$ obtained by using a multilevel affine linear preregistration;
    $\boldsymbol{u}0 \leftarrow \boldsymbol{u}_0$;
  else
    $\boldsymbol{u}0 \leftarrow I_H^h(\boldsymbol{u})$;
  end
  $\boldsymbol{u} \leftarrow$ GNIRArmijo$(\alpha, \boldsymbol{u}0)$;
end

4. Numerical experiments

In this section we present some experiments to

- compare the modeling results of our new vectorial curvature model $\mathcal{R}^{\mathrm{NewC}}$ (Algorithm 3) with the model of Chumchob-Chen-Brito [12] with regulariser $\mathcal{R}^{\mathrm{CCBc}}$ (the previously best method);
- test the performance of our proposed Algorithm 3 for $\mathcal{R}^{\mathrm{NewC}}$ with regard to parameter sensitivities.

To measure the quality of the registered images, the relative reduction of the dissimilarity rel.SSD is used, and it is defined as follows

$$\mathrm{rel} \cdot SSD = \frac{\mathcal{J}_\alpha(\boldsymbol{u})}{\mathcal{J}_{\mathrm{init}}}$$

where $\boldsymbol{u}$ is the current iteration and $\mathcal{J}_{\mathrm{init}}$ is the value of $\mathcal{J}_\alpha(\boldsymbol{u})$ at $\boldsymbol{u} = 0$. Three representative data sets (Two non-smooth registration problems and a smooth registration problem to be denoted respectively as Example 1, Example 2 and Example 3) are selected for the experiments, as shown respectively in Figure 1.

4.1. Comparison of $\mathcal{R}^{\mathrm{NewC}}$ with $\mathcal{R}^{\mathrm{CCBc}}$. In the first experiment, our aim is to investigate capabilities of $\mathcal{R}^{\mathrm{CCBc}}$ and $\mathcal{R}^{\mathrm{NewC}}$ for registration of the three test Examples $1-3$ in resolution 256×256, 512×512. Below we mainly highlight the further gains from using $\mathcal{R}^{\mathrm{NewC}}$. To be a fair comparison, we used the same Algorithm as explained in Section 3 for solving the discretised energy functional related to $\mathcal{R}^{\mathrm{CCBc}}$.

The registered results by the two models are shown in Figure 2, Figure 3 and Figure 4. On one hand, for the smooth registration problem (Example 3), one can observe that the two curvature models work fine in producing acceptable registration results, however the registered result by the new model $\mathcal{R}^{\mathrm{NewC}}$ has the more better value of rel.SSD; on the other hand, for the non-smooth registration problems (Example $1-2$), one can clearly see that our new model $\mathcal{R}^{\mathrm{NewC}}$ evidently produced visually pleasing results, especially on Example 1 with non-axis-aligned discontinuities. The main reason is that our new model utilize interdependence between the primary components of the deformation field for smooth and non-smooth registration problems.

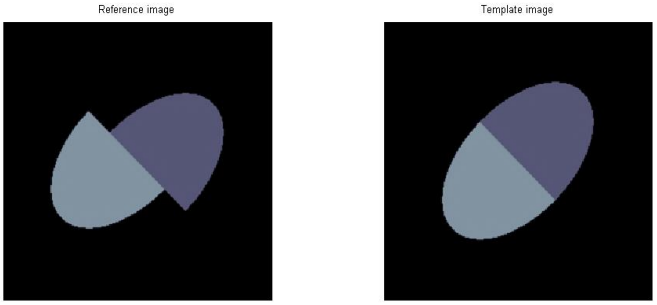
4.2. Tests of our new Algorithm 3. Here by experiments, we hope to test the convergence issues of it with regard to parameters α, β in the model and the mesh parameter h.

4.2.1. h-independent convergence tests. We shall resolve the same Example $2-3$ as above using an increasing sequence of resolutions (or a decreasing mesh parameter h) and show the results in Table 1. The results show that our new Algorithm 3 not only convergence within a very short time, but it is also accurate because the dissimilarities between the reference and registered images have been reduced more than 93%. For overall performance the experimental results suggest that our new Algorithm 3 would be preferred for practical applications.

TABLE 1. Registration results of our Algorithm 3 for processing Example $2-3$ shown respectively in Figure 1. In the table, CPU means the total runtimes including Image output and pre-registration.

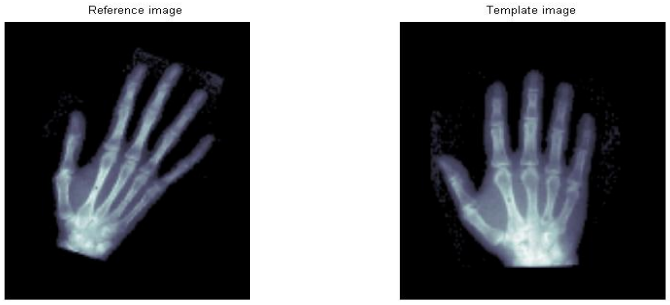
$\mathcal{R}^{\mathrm{NewC}}$ model					
Example	h	α	β	rel.SSD	CPU(second)
2	1/128	0.02	1e-6	4.1694%	9.0
	1/256	0.02	1e-6	4.3884%	21.1
	1/512	0.02	1e-6	4.2526%	68.5
Example	h	α	β	rel.SSD	CPU(second)
3	1/128	100	1e-6	0.47126%	7.2
	1/256	100	1e-6	0.46693%	13.7
	1/512	100	1e-6	0.38297%	40.8

4.2.2. α-dependence test. Here we analyze how sensitive the performance of our Algorithm 3 when varying α. To this end, our Algorithm 3 was tested on Example 3 (see Figure 1 last row) with the results shown in Table 2. Here the following parameters are used: $\beta = 1e-6$, and $h = 1/256$ for all experiments and α is varied from 1000 to 1. For this example, we can see that the performance of our Algorithm 3 is basically consistently behaved.



Example 1 (256×256)

Example 2 (512×512)



Example 3 (512×512)

FIGURE 1. Three representative data sets of registration problems. Left column: reference image R, right column: template image T. Top to bottom: Example $1-2$ (non-smooth registration problems) and Example 3 (smooth registration problem).

4.2.3. β-dependence test. As is well known, the quantities of results and the performances of some numerical schemes in solving the nonlinear system related to the total variation regularization technique are affected significantly by the value of β. Theoretically $\beta \neq 0$ should be selected to be as small as possible, thus the solution of (6) approaches to that of the original problem (2); see [1] for discussions related to another problem. Here our aim is to see how our Algorithm is affected when varying the values of β. To this end, the Algorithm 3 was tested on the non-smooth Example 2 as from Figure 1 middle row with the results shown in

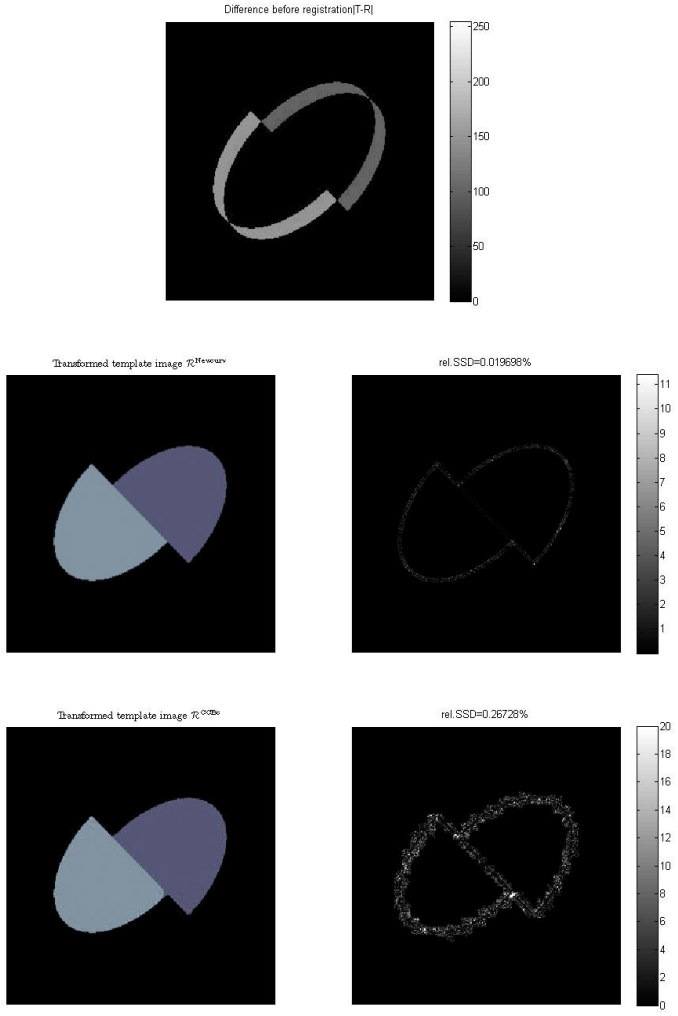


FIGURE 2. Registration results for two sliding objects of size 256×256 (Example 1). Top row: Difference between reference image and template image before registration. Middle row left: registered template image by $\mathcal{R}^{\text{NewC}}$; Middle row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{NewC}}$; Last row left: registered template image by $\mathcal{R}^{\text{CCBc}}$; Last row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{CCBc}}$. Here the heuristically best α were selected for all cases and $\beta = 10^{-6}$.

Table 3. Here the following parameters are taken: $\alpha = 0.02$, and $h = 1/256$ for all experiments and β is varied from 10^{-16} to 1. For this example, on one hand we can see our Algorithm is still convergent when β is very small; On the other hand, we can also observe the quality of registered image by Algorithm 3 is not sensitive as β reduces.

FIGURE 3. Registration results for two book images of size 512×512 (Example 2). Top row: Difference between reference image and template image before registration. Middle row left: registered template image by $\mathcal{R}^{\text{NewC}}$; Middle row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{NewC}}$; Last row left: registered template image by $\mathcal{R}^{\text{CCBc}}$; Last row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{CCBc}}$. Here the heuristically best α were selected for all cases and $\beta = 10^{-6}$.

5. Conclusions

We proposed a new vectorial curvature model for image registration in this paper, to make full use of interdependence between the primary components of the deformation field for smooth and non-smooth registration problems. The proposed regularizer is related but not identical to previous high order regularizers that have been proved to be useful in vector-valued image denoising [4, 5, 7] and in optical flow computation [2, 3]. To solve the new model, we proposed a fixed point method combined with a Gauss-Newton scheme with Armijo's Line Search and further with a multilevel method to achieve fast convergence. Numerical experiments

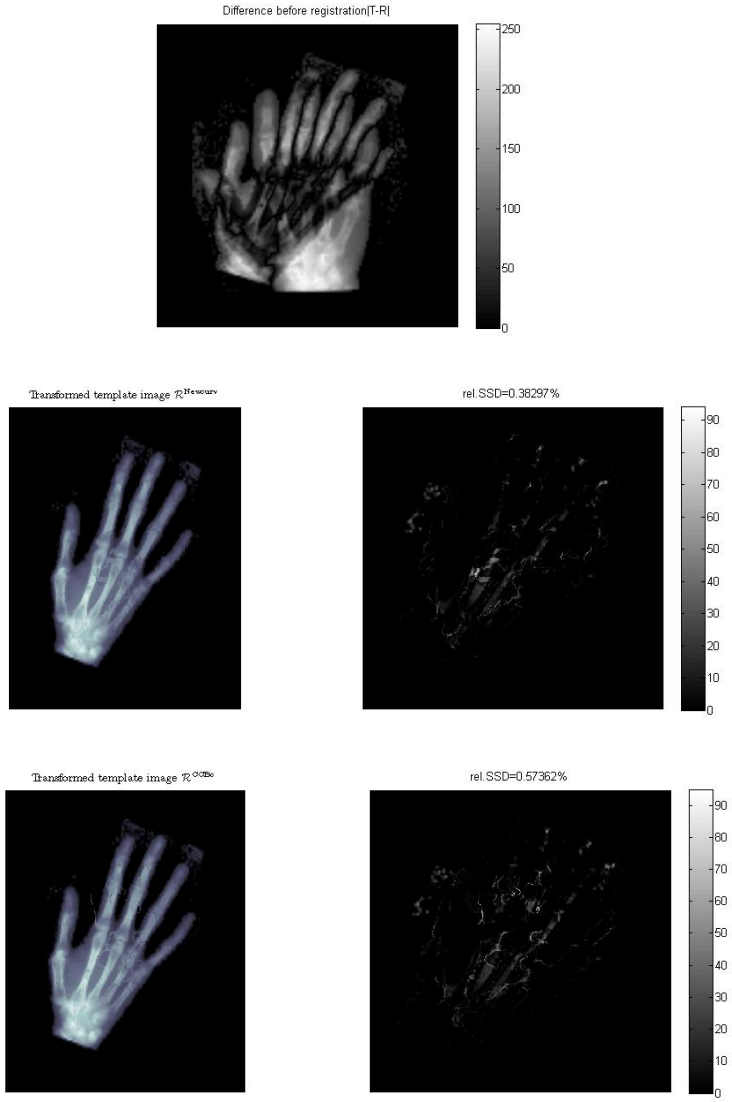


FIGURE 4. Registration results for **X**-ray images of size 512×512 (Example 3).Top row: Difference between reference image and template image before registration. Middle row left: registered template image by $\mathcal{R}^{\text{NewC}}$; Middle row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{NewC}}$; Last row left: registered template image by $\mathcal{R}^{\text{CCBc}}$; Last row right: Difference between reference image and deformed template image after registration by $\mathcal{R}^{\text{CCBc}}$. Here α were well-selected for all regularizer techniques and $\beta = 10^{-6}$ was selected.

TABLE 2. Results for α-dependence tests of Algorithm 3 for Example 3 shown in Figure 1's last row.

α	β	rel.SSD
1000	1e-6	0.59398%
500	1e-6	0.53479%
200	1e-6	0.55534%
100	1e-6	0.46693%
10	1e-6	0.48941%
1	1e-6	0.51263%

TABLE 3. Results for β-dependence tests of Algorithm 3 for Example 2 shown in Figure 1's middle row.

α	β	rel.SSD
0.02	1e-16	4.3767%
0.02	1e-12	4.3767%
0.02	1e-8	4.3846%
0.02	1e-6	4.3884%
0.02	1e-4	4.4124%
0.02	1e-2	4.4739%
0.02	1e-0	4.9293%

confirm that the proposed method can effectively find a highly accurate solution for both synthetic and realistic images and produce more robust registration results in quality than the previously best model [12]. Future works will consider mesh constraints, multi-modality registration and other generalisations.

References

[1] R.Acar and C.R.Vogel, Analysis of bounded variation penalty methods for ill-posed problems, Inverse Problems, 10 (1994), pp. 1217-1229.
[2] G. Aubert, R. Deriche, and P. Kornprobst, Computing optical flow via variational techniques, SIAM J. Appl. Math., 60:1 (1999), pp.156–182, .
[3] G. Aubert and P. Kornprobst, Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, (2nd Edition). Springer, 2006.
[4] P. Blomgren and T.F. Chan, Color TV: total variation methods for restoration of vector-valued images, IEEE Trans. Image Proc., 7 (1998), pp. 304–309.
[5] X. Bresson and T.F. Chan, Fast dual minimization of the vectorial total variation norm and applications to color image processing, Inverse Problems and Imaging, 2 (2008), pp. 455– 484.
[6] C. Brito-Loeza and K. Chen, Multigrid algorithm for high order denoising. SIAM J. Imaging Sciences, 4 (2010), pp. 363-389.
[7] C. Brito-Loeza and K. Chen, On high-order denoising models and fast algorithms for vector-valued images, IEEE Transactions on Image Processing, 19 (2010), pp. 1518–1527.
[8] L. G. Brown, A survey of image registration techniques, ACM Computing Surveys, 24 (1992), pp. 325–376.
[9] T.F. Chan, K. Chen, and J.L. Carter, Iterative methods for solving the dual formulation arising from image restoration, Electronic Transactions on Numerical Analysis, 26 (2007), pp.299–311.
[10] K. Chen and X.-C. Tai, A nonlinear multigrid method for total variation minimization from image restoration, Journal of Scientific Computing, 32:2, (2007), pp.115–138.
[11] N. Chumchob and K. Chen , A Robust Multigrid Approach for Variational Image Registration Models, Journal of Computational and Applied Mathematics , 236 (2011) 653–674.

[12] N. Chumchob, K. Chen and C. Brito-Loeza, A Fourth Order Variational Image Registration Model And Its Fast Multigrid Algorithm, SIAM J. Multiscale Modeling & Simulation, 9:1 (2011), pp. 89–128.
[13] B. Fischer and J. Modersitzki, Fast diffusion registration, Contemporary Mathematics, 313 (2002), pp. 117–129.
[14] B. Fischer, and J. Modersitzki, Curvature based Image Registration, Journal of Mathematical Imaging and Vision, 18 (2003), pp. 81–85.
[15] C. Frohn-Schauf, S. Henn, L. Hömke, and K. Witsch, Total variation based image registration, in Proceedings of the International Conference on PDE-Based Image Processing and Related Inverse Problems Series: Mathematics and Visualization, 2006, Springer-Verlag, pp. 305–323.
[16] C. Frohn-Schauf, S. Henn, and K. Witsch, Multigrid based total variation image registration, Comput. Vis. Sci., 11(2008), pp. 101–113.
[17] P.E.Gill, W. Murray and M. H. Wright, Practical optimization, Academic Press, London, 1981.
[18] G. Golub, M. Heath, and G. Wahba, Generalized cross-validation as a method for choosing a good ridge parameter, Technometrics 21 (1979), pp. 215–223.
[19] E. Haber, S. Heldmann, and J. Modersitzki, Adaptive mesh refinement for non parametric image registration, SIAM J. Sci. Comput., 30:6, (2008), pp. 3012–3027.
[20] E. Haber, S. Heldmann, and J. Modersitzki, A computational framework for image-based constrained registration, Linear Algebra and its Application, 431:3–4, (2009), pp. 459–470.
[21] E. Haber, R. Horesh, and J. Modersitzki, Numerical optimization for constrained image registration, Numer. Linear Algebra Appl. 17:2-3, (2010), pp. 343–359.
[22] E. Haber and J. Modersitzki, Numerical solutions of volume preserving image registration, Inverse Problems, 20 (2004), pp. 1621–1638.
[23] E. Haber and J. Modersitzki, *A Multilevel Method for Image Registration*, SIAM J. Sci. Comput., 27:5 (2006), pp. 1594–1607.
[24] S. Henn, A Multigrid Method for a Fourth-Order Diffusion Equation with Application to Image Processing, SIAM J. Sci. Comput., 27 No.3(2005), pp. 831-849.
[25] J. Modersitzki, Numerical Methods for Image Registration, Oxford University Press, New York, 2004.
[26] J. Modersitzki, FAIR: Flexible Algorithms for Image Registration, SIAM, Philadelphia, 2009.
[27] J.Nocedal and S.J.Wright, Numerical optimization, Springer-Verlag, New York, 1999.
[28] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys.D, 60 (1992), pp. 259–268.
[29] J. Savage and K. Chen, An improved and accelerated nonlinear multigrid method for total-variation denoising, International Journal of Computer Mathematics, 82:8, (2005), pp.1001–1015.
[30] M. Stürmer, H. Köstler, and U. Rüde, A fast full mulitigrid solver for applications in image processing, Numer. Linear Algebra Appl., 15 (2008), pp. 187–200.
[31] P. Thevenaz and M. Unser, Optimization of mutual information for multiresolution image registration, IEEE Trans. Image Process., 9 (2000), pp. 2083–2089.
[32] C. R. Vogel and M. E. Oman, Iterative methods for total variation denoising, SIAM J. Sci. Comput., 17 (1996), pp.227–238.
[33] G. Wahba, Spline Models For Observational Data, SIAM publications, Philadelphia, 1990.

School of Mathematical Sciences, Dalian University of Technology, Liaoning 116024, P R China
E-mail: zhangjindlut@163.com

Department of Mathematical Sciences and Centre for Mathematical Imaging Techniques, the University of Liverpool, United Kingdom (For Correspondence).
E-mail: k.chen@liverpool.ac.uk
URL: http://www.liv.ac.uk/cmit, http://www.liv.ac.uk/∼cmchenke