A novel few-shot learning framework for supervised diffeomorphic image registration network

Ke Chen, **Huan Han**, **Junping Wei** and **Yimin Zhang**

***Abstract*— Image registration is a key technique in image processing and analysis. Due to its high complexity, the traditional registration frameworks often fail to meet real-time demands in practice. To address the real-time demand, several deep learning networks for registration have been proposed, including the supervised and the unsupervised networks. Unsupervised networks rely on large amounts of training data to minimize specific loss functions, but the lack of physical information constraints results in the lower accuracy compared with the supervised networks. However, the supervised networks in medical image registration face two major challenges: physical mesh folding and the scarcity of labeled training data. To address these two challenges, we propose a novel few-shot learning framework for image registration. The framework contains two parts: random diffeomorphism generator (RDG) and a supervised few-shot learning network for image registration. By randomly generating a complex vector field, the RDG produces a series of diffeomorphism. With the help of diffeomorphism generated by RDG, one can use only a few image data (theoretically, one image data is enough) to generate a series of labels for training the supervised few-shot learning network. Concerning the elimination of the physical mesh folding phenomenon, in the proposed network, the loss function is only required to ensure the smoothness of deformation (no other control for mesh folding elimination is necessary). The experimental results indicate that the proposed method demonstrates superior performance in eliminating physical mesh folding when compared to other existing learning-based methods. Our code is available at this link https://github.com/weijunping111/RDG-TMI.git**

***Index Terms*— image registration, Beltrami coefficient, diffeomorphism, few-shot learning**

I. Introduction

This work was supported by the National Key Research and Development Program of China (2020Y-FA0714200), the National Natural Science Foundation of China (No. 12171379, 12271417 and 12471484) and the Fundamental Research Funds for the Central Universities, China. Corresponding authors: Huan Han and Yimin Zhang.

K. Chen is with Department of Mathematics and Statistics, University of Strathclyde, Glasgow, G1 1XH, UK (e-mail: K.Chen@strath.ac.uk), OrCiD link: https://orcid.org/0000-0002-6093-6623.
H. Han is with the Department of Mathematics, Wuhan University of Technology, Wuhan 430070, China, (e-mail: hanhuan11@whut.edu.cn), OrCiD link:https://orcid.org/0000-0002-6334-4738.
J. Wei is with the Center for Mathematical Sciences, Wuhan University of Technology, Wuhan 430070, China, (e-mail: weijunpingacca@163.com), OrCiD link:https://orcid.org/0009-0008-2781-2160.
Y. Zhang is with the Center for Mathematical Sciences, Wuhan University of Technology, Wuhan 430070, China, (e-mail: zhangyimin@whut.edu.cn), OrCiD link:https://orcid.org/0000-0002-2617-8574.
All the authors contributed equally to this work.

IMAGE registration is a challenging task in image processing and analysis. Generally speaking, the goal of image registration is to establish a spatial relationship between the images acquired in different time, from different spaces or with different devices. Specifically, for some bounded domain $\Omega \subseteq \mathbb{R}^d (d = 2, 3)$ and two images $T, R : \Omega \to \mathbb{R}$, the goal of image registration is to find a spatial transformation $\varphi : \Omega \to \Omega$ such that the deformed floating image $T \circ \varphi(\cdot)$ looks like the target image $R(\cdot)$ as much as possible. It is an ill-posed problem, and to overcome the ill-poseness, one classical way is to add some regularization or constraints on φ. Therefore, the variational framework for image registration is formulated as follows:

$$\min_{\varphi \in \mathcal{K}} \mathcal{D}(T \circ \varphi, R) + \mathcal{S}(\varphi), \tag{1}$$

where $\mathcal{K}$ is some proper function space, $\mathcal{D}(T \circ \varphi, R)$ is the fitting term measuring the similarity between $T \circ \varphi$ and R, $\mathcal{S}(\varphi)$ is regularization to eliminate unexpected solutions.

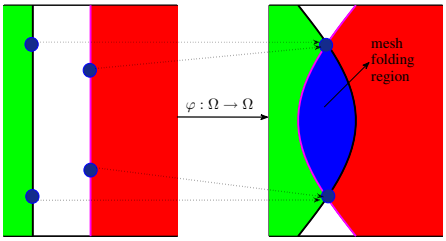


Fig. 1: Physical mesh folding phenomenon caused by the unexpected deformation φ. The left image illustrates the location of three features along with four pixels before transformation, where the green, white, red colors denote the features. The right image denotes the new locations of the features after being wrapped by the specific transformation $\varphi : \Omega \to \Omega$, where the blue part denotes the physical mesh folding region.

Without loss of generality, we mainly focus on the discussion of 2D ($d = 2$) image registration in this paper. Within the framework (1), several pioneering models are proposed, for instance, Demons model [1], Vectorical total variation (VTV) model [2] and Bounded deformation (BD) model [3]. However, physical mesh folding is not taken into consideration in these models. Taking Fig 1 as an example, with an assumption that the green and red region denote two different organs, respectively, and φ transforms the left part

to the right part, then one can notice that two different organs are mixed together (blue region) by the given transformation. This is so called physical mesh folding phenomenon, which is not allowed in medical image registration. To eliminate mesh folding, several diffeomorphic registration frameworks which search for a C^1 continuous and invertible mapping φ are proposed, for example, the Inverse-consistent model [4], Quasi conformal/conformal models [5]–[7], Positive Jacobian determinant constrained model [8], the Ogden material stored energy model [9] and the LDDMM framework [10]. These models are all based on the traditional variational framework, which provides more interpretable solutions and does not rely on the large amount of training data. However, due to its high complexity, the traditional variational registration frameworks could not satisfy the real-time demand in practice. To address the real-time demand, several deep learning networks [11]–[15] for registration are proposed, including unsupervised networks and supervised networks. In unsupervised learning network [11], [13], [14], the training of the network is driven by the similarity $\mathcal{D}(T \circ \varphi, R)$ between $T \circ \varphi$ and R, and the general form of the loss function is formulated as

$$\mathcal{L}_{loss}(\varphi) = \mathcal{D}(T \circ \varphi, R) + \lambda \mathcal{L}_{reg}(\varphi), \tag{2}$$

where $\mathcal{L}_{reg}(\varphi)$ is regularization on φ, and λ is a positive number to balance these two terms. The issue to be addressed is how to obtain a non-folding transformation φ as effectively as the above-mentioned diffeomorphic registration models.

Following from the structure of $\mathcal{L}_{loss}(\varphi)$ in Eq. (2), one can notice that the training of the unsupervised network is driven by a large amount of data automatically without setting any prior information on the solution. This may produce some unexpected solutions because there is no uniqueness of solution for the registration framework. To ensure that the solutions satisfy some specific demand in application, several supervised networks [12] are trained by minimizing the following loss function:

$$\mathcal{L}_{loss}(\varphi) = \mathcal{D}(T \circ \varphi, R) + \mathcal{L}_{sim}(\varphi, \varphi_r) + \lambda \mathcal{L}_{reg}(\varphi), \tag{3}$$

where φ_r is the deformation field, and the minimization of $\mathcal{L}_{sim}(\varphi, \varphi_r)$ drives the network to be trained to produce a solution φ similar to φ_r.

The overview of supervised and unsupervised networks is presented in Section 2. Evidently, there is a strong need for new methods to address the mesh folding problem. For the supervised networks in medical image registration, they face two challenges: physical mesh folding elimination and no labels φ_r for training. To address these two challenges, in this paper, we propose a novel few-shot learning framework for diffeomorphic image registration. The framework contains two parts: random diffeomorphism generator (RDG) and a supervised few-shot learning network for diffeomorphic image registration. By randomly generating the Beltrami coefficients (complex vertor field), the RDG produces a series of diffeomorphsim. With the help of diffeomorphsim generated by RDG, one can use only a few image (theoretically, one image is enough) to generate a series of diffeomorphic labels for training the supervised few-shot learning network.

The contribution of this paper contains:

- Propose a novel variational model (RDG (10)) to generate diffeomorphism via randomly generated complex field $\mu : \Omega \to \mathbb{C}$;
- Propose a fast multi-grid (MG) method for the numerical implementation of the proposed model;
- Design a few shot learning network for supervised diffeomorphic image registration, using the labels generated by RDG (10).

The paper's overall framework is presented in Figure 2.

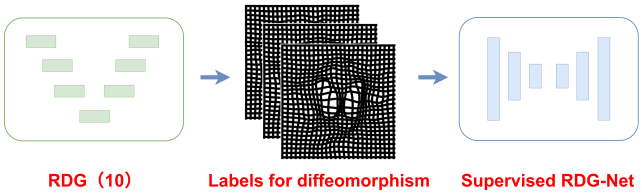


Fig. 2: Flowchart of the proposed few-shot learning framework for supervised diffeomorphic image registration network: Step 1. RDG (10) generates a large amount of labels φ^j using randomly generated complex vector field $\tilde{\mu}$; Step 2. Using the input image T_i to produce $R_{ij} = T_i(\varphi^j)$ and labels (T_i, R_{ij}, φ^j). Step 3. Using the labels (T_i, R_{ij}, φ^j) to train the supervised RDG-Net.

The remaining sections are organized as follows: In Section II, the related work of diffeomorphic image registration is reviewed. In Section III, we propose a novel variational model to generate diffeomorphism via randomly generated complex fields. Based on the labels generated by RDG (10), we propose a supervised few-shot learning framework for diffeomorphic image registration. In Section IV, several numerical tests are performed to validate the results. In Section V, we conclude our work and list some problems for future research.

II. Related Work

A. Traditional image registration methods

Traditional image registration formulates the registration as a variational model, for example, the framework (1). In the pioneering work, many classical models [1]–[3] are proposed. In these early works, physical mesh folding phenomenon has not drawn much attention. As it is stated in Fig 1 in Section 1, mesh folding elimination is a key and challenging task in medical image registration. From Fig 1, one can notice that the essential reason for the appearance of mesh folding (Blue region) is attributed to the fact that the transformation $\varphi : \Omega \to \Omega$ is not a one-to-one mapping. This observation motivates the need for constraints to ensure that the final solution φ of Eq. (1) is a one-to-one mapping. For this purpose, Zeng and Chen [4] introduced the inverse-consistent model by simultaneously using two different variational models which produce two mappings (one is just the inverse of the other). In another view, the Inverse Function Theorem [16] provides a sufficient condition to ensure that φ is a one-to-one mapping. In this framework, the Jacobian determinant of φ is necessary to be greater than zero. Based on this condition, Zhang et al. reformulated the framework (1) by restricting φ to the set

whose Jacobian determinant is greater than 0 everywhere. Analternative, Lui et at [5], [7], [17], [18] used the Quasi-conformal theory to transform the conditions to its equivalent form which restrict φ to the set whose infinite norm of Beltrami coefficient (see subsection B in Section III for details) is less than 1. As a simplification, Han et al [6], [19]–[21] proposed a relaxed conformal framework, which provides a much simple structure for numerical implementation. Furthermore, Debroux et al. [9] introduced a penalty term to penalize the voxel whose Jacobian determinant is less than 0. Though these models achieve satisfactory results in mesh folding elimination and get accurate results, the high complexity of their algorithms limit real-time applicability in scenarios like surgical guidance.

B. Deep learning-based image registration methods

To reduce the complexity of traditional registration framework, several deep learning networks [11]–[14] for registration are proposed in recent years. These networks can be classified into unsupervised networks [11], [13], [14] and supervised networks [11], [13], [14] determined by whether the training of the network needs data labels φ_r. As it is stated in Section 1, the training of the unsupervised network is driven by a large amount of data to minimize the loss function (2), and the quality of the output solution is determined by some specific structures in the network. For example, the Convolutional Neural Networks (CNNs). Among the CNNs, the most famous is the U-Net architecture which incorporates down-sampling and symmetric up-sampling modules to capture information across different scales of receptive fields. Taking U-Net as one part of the structure of the network, the Spatial Transformer Network (STN) [14], [22] is a network architecture that performs transformations across the entire image by predicting transformation parameters. This includes operations such as translation, rotation, stretching, scaling, and non-rigid deformations. The core of STN is a localization network that takes the images as input and outputs the parameters for spatial transformations, such as six parameters for affine transformations or eight parameters for plane projective transformations. In contrast to STN, VoxelMorph (VM) [11] is an unsupervised network that achieves registration at the pixel level. Following VM, several variants of it have emerged, further enhancing its capabilities in image registration tasks. Combing STN and VM, Huang [13] proposed an unsupervised coarse-to-fine registration framework and introduced dual consistency constraint to eliminate the physical mesh folding. Chen, Li and Lui [12] also proposed a learning framework for diffeomorphic image registration based on quasi-conformal theory (QCRegNet). This is a two-stage neural network that includes an estimator network and a Beltrami solver network. The estimator network takes an image pair as input and outputs the Beltrami coefficient. Subsequently, the Beltrami solver network utilizes the Beltrami coefficient obtained from the estimator to reconstruct the corresponding quasiconformal mapping. In addition, there are also some diffusion models [24] based registration networks, for example, DiffuseReg network [25] and FSDiffReg network [26]. For an overview of deep learning based medical image registration, the readers are refer to [27].

These models are all unsupervised, which implies that during the training process, we are unaware of the labels. Therefore, if the actual deformation fields were known, the network could be trained more effectively in supervised learning network for image registration. For the supervised learning network, the main difficulty is the lack of labels φ_r. To address this difficulty, there are two kinds of approaches reported. One is to obtain the real labels φ_r by iteratively solving the traditional registration models, which needs too much CPU consumption. The other way is to randomly generate six parameters (3 parameters for rotation, and 3 parameters for translation) to produce a series of affine transformations, for example, the AIRNet in [28]. Note that here the output of AIRNet [28] is an affine transformation, which can be simulated by six parameters. For the supervised network of diffeomorphic image registration, it is necessary to output a C^1 continuous one-to-one mapping, which belongs to an infinite dimensional space. Due to this fact, infinite parameters are needed to simulate some specific diffeomorphism. This leads a challenge in generation of diffeomorphism. How to address this challenge in supervised learning network for diffeomorphic image registration, is a problem to be addressed in this paper.

III. Method

In this section, we propose a novel few-shot learning framework for diffeomorphic image registration. The proposed framework contains three parts:

- **The random diffeomorphism generator (RDG).** The proposed variational model inverts the real labels φ_r^j via the information provided by the randomly generated complex vector fields $\mu^j : \Omega \to \mathbb{C}$ with $\|\mu^j\|_\infty < 1$ (Note that here $\|\mu^j\|_\infty = \max_{\mathbf{x}\in\Omega} |\mu^j(\mathbf{x})|$);
- **Network training data generator.** Using the labels φ_r^j $(j = 1, 2, \cdots)$ provided by RDG, and the image T_i $(i = 1, 2, \cdots)$ from the image dataset to generate the training labels $(T_i, R_{i,j}, \varphi_r^j)$ (where $R_{i,j} = T_i(\varphi_r^j)$) for the diffeomorphic registration network;
- **Supervised learning network for diffeomorphic image registration.** Using the labels $(T_i, R_{i,j}, \varphi_r^j)$ provided by the RDG, design a few-shot learning network for medical image registration.

A. The random diffeomorphism generator

As stated at the end of Section II, the key to randomly generate the diffeomorphism lies in establishing the relationship between the randomly generated numbers and the diffeomorphism. Motivated by the Quasi-conformal theory [5] in traditional registration framework, by modeling the transformation $\varphi = (u, v)^T$ as a complex function $\varphi(\mathbf{x}) = u(\mathbf{x}) + iv(\mathbf{x})$ (i is imaginary unit), a complex field $\mu : \Omega \to \mathbb{C}$ is generated by

$$\mu(\varphi(\mathbf{x})) = \frac{(\partial_x u - \partial_y v) + i(\partial_x v + \partial_y u)}{(\partial_x u + \partial_y v) - i(\partial_x u - \partial_y v)}, \tag{4}$$

where μ is the so called Beltrami coefficient of $\boldsymbol{\varphi}$.

Through straightforward computation, it is concluded that [5] $\|\mu\|_\infty < 1$ is equivalent to $\det(\nabla\boldsymbol{\varphi}) > 0$ (a sufficient condition to ensure that $\boldsymbol{\varphi}$ is a one-to-one mapping). Therefore, the combination of Eq. (4) with the condition $\|\mu\|_\infty < 1$ establishes the relationship between the complex vector field $\mu : \Omega \to \mathbb{C}$ and diffeomorphism $\boldsymbol{\varphi} : \Omega \to \Omega$.

Obviously, the direct problem (computing μ via the information provided by $\boldsymbol{\varphi}$) is easy. However, the goal of this section is to generate diffeomorphism $\boldsymbol{\varphi}$ by using the information of μ (the so-called inverse problem). Motivated by the modeling of inverse problem, we propose the following variational framework for generating diffeomorphism:

$$\boldsymbol{\varphi} = \arg\min \lambda\mathcal{F}(\boldsymbol{\varphi}) + \alpha\mathcal{G}(\boldsymbol{\varphi}), \tag{5}$$

here and in what follows, $\rho, \tau : \Omega \to \mathbb{R}$, $\tilde{\mu}(\mathbf{x}) = \rho(\mathbf{x}) + i\tau(\mathbf{x})$ is a given complex field with $\|\tilde{\mu}\|_\infty < 1$, $\lambda, \alpha > 0$, the fitting term

$$\mathcal{F}(\boldsymbol{\varphi}) = \int_\Omega |\tilde{\mu} - \mu(\boldsymbol{\varphi})|^2 d\mathbf{x}, \tag{6}$$

and the regularization

$$\mathcal{G}(\boldsymbol{\varphi}) = \int_\Omega (\frac{\partial\varphi_1}{\partial x_1} - \frac{\partial\varphi_2}{\partial x_2})^2 + (\frac{\partial\varphi_1}{\partial x_2} + \frac{\partial\varphi_2}{\partial x_1})^2 d\mathbf{x}. \tag{7}$$

Remark 1: In [12], a partial differential equation (PDE) solver for network is proposed by minimizing the fitting term $\mathcal{F}(\boldsymbol{\varphi})$. This is an ill-posed problem, which may have two shortages: many unexpected solutions and the solution is not smooth (see Fig 5 for later comparison in Section IV). To address the above problems, the Cauchy-Riemann regularization [6] on $\mathcal{G}(\boldsymbol{\varphi})$ is used to overcome the ill-poseness of the original minimization problem. It has two advantages: (I) restricting $\boldsymbol{\varphi}$ to a diffeomorphism to eliminate mesh folding phenomenon; (II) providing smoothness prior to the solution $\boldsymbol{\varphi}$. We further note that directly generating random $\boldsymbol{\varphi}$ (without μ) is impractical, as it is inherently difficult to satisfy both diffeomorphism and smoothness requirements simultaneously.

Substitute Eq. (4) into Eq. (6), there holds

$$|\tilde{\mu} - \mu(\boldsymbol{\varphi})|^2 = \left|\frac{(\rho + i\tau)\cdot(A + iB) - (C + iD)}{A + iB}\right|^2, \tag{8}$$

where

$$A = (u_{x_1} + v_{x_2}), B = (v_{x_1} - u_{x_2}),$$
$$C = (u_{x_1} - v_{x_2}), D = (u_{x_2} + v_{x_1}).$$

Note that the denominator $A + iB$ in Eq. (8) has the possibility to be 0, which may lead the singularity in variational framework (5). To address the singularity, we reformulate the fitting term to the following equivalent form:

$$\begin{aligned}\bar{\mathcal{F}}(\boldsymbol{\varphi}) &= \int_\Omega |(\rho + i\tau)(A + iB) - (C + iD)|^2 d\mathbf{x} \\ &= \int_\Omega |\,[(\rho-1)u_{x_1} - (\rho+1)v_{x_2} - \tau B] \\ &\quad + i\,[(\rho-1)v_{x_1} - (\rho+1)u_{x_2} + \tau A]\,|^2 d\mathbf{x}.\end{aligned} \tag{9}$$

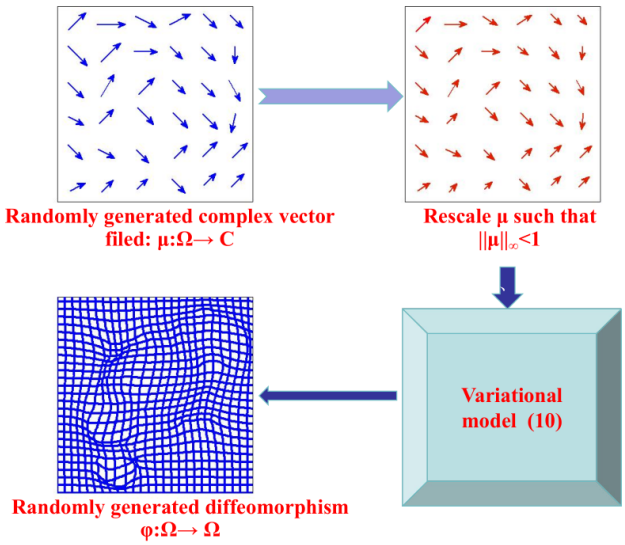


Fig. 3: Flowchart of the RDG contains four steps: (a) Randomly generating a complex vector field $\tilde{\mu} : \Omega \to \mathbb{C}$; (b) Rescaling the complex vector field $\tilde{\mu} : \Omega \to \mathbb{C}$ such that $\|\tilde{\mu}\|_\infty < 1$; (c) Solving the solution of Eq. (10); (d) Generating the diffeomorphism $\boldsymbol{\varphi} : \Omega \to \Omega$.

Using these equivalent forms, we solve the following variational model for generating the diffeomorphism in numerical implementation:

$$\boldsymbol{\varphi} = \arg\min \lambda\bar{\mathcal{F}}(\boldsymbol{\varphi}) + \alpha\mathcal{G}(\boldsymbol{\varphi}). \tag{10}$$

That is, by randomly generating a complex field $\tilde{\mu} : \Omega \to \mathbb{C}$ with $\|\tilde{\mu}\|_\infty < 1$, one can use the solution of Eq. (10) to invert diffeomorphism $\boldsymbol{\varphi}$. The flow of this process is listed on Fig 3.

Remark 2: RDG synthesize diffeomorphic labels via quais-conformal theory maintaining architecture independence. In contrast, the architecture-based methods [29], [30] inherently integrate deformation modeling into network pipelines, risking topology violations, for example, the physical mesh folding still exists in unsupervised networks [29], [30] though specific structures are designed to embed the stationary velocity field (SVF). Therefore, the RDG based registration network is more flexible and robust for diffeomorphic image registration.

B. Numerical solver of RDG framework (10)

The existence of solution of variational framework (10) can be obtained by proving the lower weak semi-continuity of the functional $\bar{\mathcal{F}}(\boldsymbol{\varphi}) + \alpha\mathcal{G}(\boldsymbol{\varphi})$. The technique is similar to [6]. Here we do not repeat it.

By variational theory [16], the solution of Eq. (10) satisfies the following PDE:

$$\begin{cases} f_1(u,v) - \triangle u = 0, \\ f_2(u,v) - \triangle v = 0, \end{cases} \tag{11}$$

where $f_1(u,v) = u_{x_1}[P_{x_1} + (-2\tau)_{x_2}] + u_{x_2}[Q_{x_2} + (-2\tau)_{x_1}] + v_{x_1}Z_{x_2} - v_{x_2}Z_{x_1}$, $f_2(u,v) = v_{x_1}[P_{x_1} + (-2\tau)_{x_2}] + v_{x_2}[Q_{x_2} +$

$(-2\tau)_{x_1}] + u_{x_2}Z_{x_1} - u_{x_1}Z_{x_2}$ and $P = (\rho-1)^2 + \tau^2, Q = (\rho+1)^2 + \tau^2, Z = 1 - \rho^2 - \tau^2$.

Eq. (11) is a nonlinear PDE, whose numerical implementation is faced with stringent conditions for the convergence of the algorithm. To ensure the robustness of the algorithm and make it gradually converge to the final solution, we introduce a time variable t and reformulate u and v to be functions of $\mathbf{x}$ and t. That is, $u = u(\mathbf{x}, t)$ and $v = v(\mathbf{x}, t)$. Based on this reformulation and gradient flow approach [19], Eq. (11) becomes:

$$\begin{cases} \dfrac{\partial u}{\partial t} = \triangle u - f_1(u, v), \\ \dfrac{\partial v}{\partial t} = \triangle v - f_2(u, v), \end{cases} \tag{12}$$

It is easy to show that the steady state solution ($t \to +\infty$) of Eq. (12) is also the solution of original problem (11).

Next, we focus on the numerical implementation of Eq. (12). The image grid is

$$\Omega = \{\mathbf{x} : \mathbf{x} = (i, j),\ 1 \le i \le M,\ 1 \le j \le N\}, \tag{13}$$

where (M, N) denote the number of rows and columns, respectively, and the time grid is

$$[0, T_m] = \{t : t_n = n,\ 1 \le n \le P\}, \tag{14}$$

with $\delta = \frac{T_m}{P}$.

With these notations, $g_{i,j} = g(i, j)$ denotes the intensity of g (here $g = T$ or R) at position $\mathbf{x}$. In discretization settings, it is involved with the numerical approximation of differential operator $\frac{\partial}{\partial t}$ and Δ. For function g ($g = u, v$), there holds,

$$\left(\frac{\partial g}{\partial t}\right)_{(\mathbf{x}, t_n)} = \frac{g_{i,j}^{n+1} - g_{i,j}^{n}}{\delta} \tag{15}$$

and

$$(\Delta g)_{(\mathbf{x}, t_n)} = g_{i+1,j}^{n} + g_{i-1,j}^{n} + g_{i,j+1}^{n} + g_{i,j-1}^{n} - 4g_{i,j}^{n}. \tag{16}$$

Using Eq. (15)-(16), Eq. (12) is discretized into an algebraic system. The two PDEs in Eq. (12) are of the same type, here we take the first PDE on u as an example to illustrate the computation process. Note that the numerical implementation is performed in two dimensional region Ω. To reduce the complexity of the algorithm, motivated by the alternative direction impilicit (ADI) scheme [19], we add a time node $n + \frac{1}{2}$ between n and $n+1$, and split the computation process from n to $n+1$ into two 1D problems:

$$\begin{cases} \dfrac{u^{n+\frac{1}{2}} - u^{n}}{\delta/2} = \triangle u^{n+\frac{1}{2}} - f_1(u^{n}, v^{n}), \\ \dfrac{u^{n+1} - u^{n+\frac{1}{2}}}{\delta/2} = \triangle u^{n+1} - f_1(u^{n+\frac{1}{2}}, v^{n+\frac{1}{2}}). \end{cases} \tag{17}$$

With the notations in Eq. (15)-(16), Eq. (17) is reformulated into the algebraic systems of the following type:

$$\begin{cases} L_1 u^{n+\frac{1}{2}} = h_1^{n}, \\ L_2 u^{n+1} = h_2^{n+\frac{1}{2}}, \end{cases} \tag{18}$$

where L_1 and L_2 are two 1D difference operators, and h_1^n, $h_2^{n+\frac{1}{2}}$ are functions of u^n and $u^{n+\frac{1}{2}}$, respectively.

In this paper, we use the multi-grid method [19], [31] to solve each equation in Eq. (18). It is essentially an iteration scheme of several V-Cycles. For each V-cycle, the computation process is listed as the following four Steps (taking the numerical implementation of $L_1 u^{n+\frac{1}{2}} = h_1^n$ as an example):

By down-sampling the region Ω at different levels, we obtain a series of grids $\{\Omega_m\}$ ($m = 0, 1, \cdots, Q$). $\Omega_Q = \Omega$ is the finest grid and Ω_0 is the coarsest grid (smallest amount of voxel).

Step 1. Smooth relaxation. Start with some initial value on Ω_Q, from Eq. (18), we have the following linear system:

$$L_{1,Q} u_Q = h_{1,Q}. \tag{19}$$

Relaxing Eq. (19) ν times using the Jacobi method [19], we obtain the smooth approximation $\bar{u}_Q$. Using $\bar{u}_Q$, the residual error is computed by :

$$r_Q = h_{1,Q} - L_{1,Q}\bar{u}_Q. \tag{20}$$

Step 2. Restriction. Restrict r_t on Ω_t to r_{t-1} on Ω_{t-1} by restriction ($t = Q, Q-1, \cdots, 1$) :

$$r_{t-1} = I_t^{t-1} r_t,$$

where I_t^{t-1} is a restriction operator from Ω_t to Ω_{t-1}.

Similar to Step 1, we solve the following equation on Ω_{t-1}:

$$L_{1,t-1} u_{t-1} = r_{t-1}, \tag{21}$$

by relaxing Eq. (21) ν times to obtain $\bar{u}_{t-1}$, and update the residual error on Ω_{t-1} by:

$$r_{t-1} = r_{t-1} - L_{1,t-1}\bar{u}_{t-1}.$$

Repeat Step 2 until the V-cycle reaches the coarsest grid Ω_0 ($t = 1$).

Step 3. Solution on coarsest grid Ω_0. On coarsest grid Ω_0, the system contains only a small number of unknown variables, making it straightforward to compute an accurate solution u_0 by the following equations:

$$L_{1,0} u_0 = r_0.$$

Step 4. Extension and correction. Following from the above process, we obtain different levels of approximation $\bar{u}_M, \bar{u}_{M-1}, ..., \bar{u}_1, u_0$ on different grids.

Start with $\tilde{u}_{t-1}$ on Ω_{t-1} ($\tilde{u}_0 = \bar{u}_0$ on Ω_0), we can interpolate $\tilde{u}_{t-1}$ onto Ω_t by:

$$\tilde{u}_t = K_{t-1}^{t} \tilde{u}_{t-1}, \tag{22}$$

where K_{t-1}^t is interpolation operator from Ω_{t-1} to Ω_t. Then the approximation on Ω_t is updated by $\tilde{u}_t = \tilde{u}_t + \bar{u}_t$.

Similar to Step 1, we use $\tilde{u}_t$ as initial value to relax Eq. (21) ν times to obtain a more accurate $\tilde{u}_t$, and Repeat Step 4 until it reaches to the finest grid Ω_Q.

At last, taking $\tilde{u}_Q$ as initial guess to relax Eq. (19) to obtain the approximation of V-cycle. The computation process of the V-cycle is listed on Fig 4.

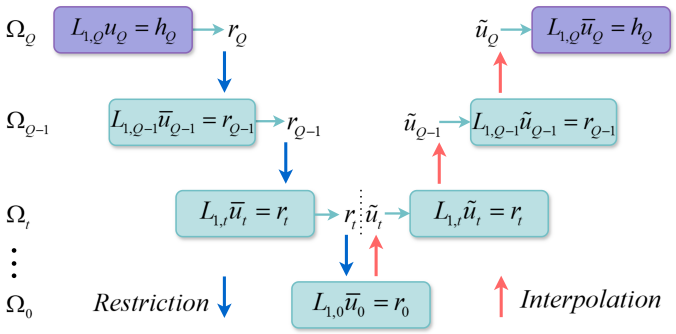


Fig. 4: V-cycle of $L_1 u^{n+\frac{1}{2}} = h_1^n$. This algorithm transfers error correction information across multigrid hierarchy through restriction (the blue arrow), coarse-grid solver $L_1\bar{u}_0 = r_0$, and interpolation (the red arrow), enabling efficient solutions to linear systems $L_1 u^{n+\frac{1}{2}} = h_1^n$, where $\Omega_i (i = 0, 1, \cdots, Q)$ denotes different levels of the regions (the larger i, the finer region) for solving the linear equations.

To validate the accuracy of RDG (10), we collect some deformations $\boldsymbol{\varphi}_{real}$ (see the first column of Fig 5) and use Eq. (4) to compute the Beltrami coefficient of $\mu(\varphi_{real})$ (see the second column of Fig 5). By using the above multigrid method to solve the inverse problem (10), we obtain the predicted deformation $\boldsymbol{\varphi}_{predict}$ (see the third column of Fig 5). From Fig 5, one can notice that the RDG has the ability to recovery the deformation by setting specific complex vector field $\tilde{\mu} : \Omega \to \mathbb{C}$. This provides theoretical support for the latter application of RDG in few-shot learning network in subsection C.

C. Label generation for the few-shot learning network

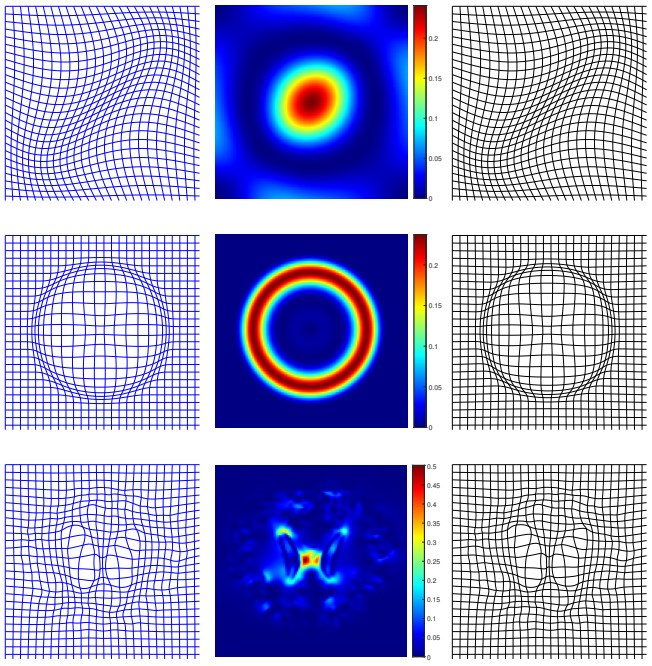
To make the diffeomorphic labels generated by RDG compatible with the applications in practice, we collect several diffeomorphism $\boldsymbol{\varphi}_{real}$ from the real data and compute the corresponding $\bar{\mu}(\boldsymbol{\varphi}_{real})$. For each $\bar{\mu}(\boldsymbol{\varphi}_{real})$, set a global small perturbation μ_1 and local large perturbation μ_2, then a series of complex field $\tilde{\mu} : \Omega \to \mathbb{C}$ is obtained by

$$\tilde{\mu} = \bar{\mu} + \mu_1 + \mu_2. \tag{23}$$

Remark 3: The random fields μ_1, μ_2 are sampled in the following way (Fig 6):
(i). Setting $\mu_1 = \rho_1 + i\tau_1$, where $\rho_1 : \Omega \to [-0.5, 0.5]$ and $\tau_1 : \Omega \to [-0.5, 0.5]$ are generated by two random matrices whose elements are restricted into the interval $[-0.5, 0.5]$, respectively, then we use Gaussian filter to convolute ρ_1 and τ_1 to achieve a smooth version of the field ρ_1 and τ_1;
(ii). Setting $\mu_2 = \rho_2 + i\tau_2$, then $\rho_2 : \Omega \to \mathbb{R}$ and $\tau_2 : \Omega \to \mathbb{R}$ are generated by restricting ρ_2 and τ_2 into a 2D Gaussian filter distribution with randomly generated expectation and variance.

By rescaling $\tilde{\mu}$ to ensure that $\|\tilde{\mu}\|_\infty < 1$, and setting $\tilde{\mu}$ as input of the RDG framework (10), we obtain a series of diffeomorphism $\boldsymbol{\varphi}_r^j$ $(j = 1, 2, \cdots)$.

Using the diffeomorphism $\boldsymbol{\varphi}_r^j$ $(j = 1, 2, \cdots)$ provided by RDG framework, we wrap each image T_i $(i = 1, 2, \cdots)$ from the dataset to create labeled tuples $(T_i, R_{i,j}, \boldsymbol{\varphi}_r^j)$ (where $R_{i,j} = T_i(\boldsymbol{\varphi}_r^j)$) . Fig 7 shows the complete flowchart for generating these diffeomorphic labels for the few-shot learning network.



(a) $\boldsymbol{\varphi}_{real}$ (b) $|\mu(\boldsymbol{\varphi}_{real})|$ (c) $\boldsymbol{\varphi}_{predcit}$

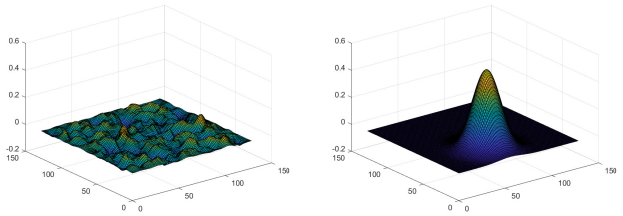
Fig. 5: The first column is some real deformations $\boldsymbol{\varphi}_{real}$; The second column is the $|\mu(\boldsymbol{\varphi}_{real})|$; The third column is deformations $\boldsymbol{\varphi}_{predcit}$ through solving the inverse problem (10) by replacing $\tilde{\mu} = \rho + i\tau$ in Eq. (10) with $\mu(\boldsymbol{\varphi}_{real}) = \rho_{real} + i\tau_{real}$. Taking the first row as an example, the first image is the mesh of a diffeomorphism $\boldsymbol{\varphi}_{real}$ coming from real data. By Eq. (4), we can compute the complex vector field $\mu(\boldsymbol{\varphi}_{real})$. The second image shows the modulus $|\mu(\boldsymbol{\varphi}_{real})|$. The third image is the mesh of the deformation $\boldsymbol{\varphi}_{predcit}$ via solving the RDG (10) by replacing $\tilde{\mu} = \rho + i\tau$ with $\mu(\boldsymbol{\varphi}_{real}) = \rho_{real} + i\tau_{real}$.

D. Supervised few-shot learning framework for diffeomorphic image registration

In this paper, we mainly focus on establishing a few-shot learning framework. Therefore, a simple structure is designed in our network. In fact, for the specific applications, special structures of the network can also be embedded into the proposed framework (for example, the Decoder-only image registration network [23]). The structure of the proposed few-shot learning framework is listed on Fig 8.

The loss function of the framework contains the following three parts:

Sum of squared difference (SSD): This paper mainly discusses the mono-modality image registration which indicates that T and R are captured by the same sensors (i.e., CT-CT, MRI-MRI). For mono-modality image registration, the most robust similarity is the sum of squared intensity difference



(a) ρ_1 (b) ρ_2

(c) τ_1 (d) τ_2

Fig. 6: Example of randomly generated vector fields $\rho_1, \tau_1, \rho_2, \tau_2$. $\rho_1 : \Omega \to [-0.5, 0.5]$ and $\tau_1 : \Omega \to [-0.5, 0.5]$ are generated by two random matrices whose elements are restricted into the interval $[-0.5, 0.5]$, respectively. $\rho_2 : \Omega \to \mathbb{R}$ and $\tau_2 : \Omega \to \mathbb{R}$ are generated by restricting ρ_2 and τ_2 into a 2D Gaussian filter distribution with randomly generated expectation and variance.

between image $T \circ \boldsymbol{\varphi}$ and R defined by

$$SSD(T, R, \boldsymbol{\varphi}) = \int_{\Omega} (R - T \circ \boldsymbol{\varphi})^2 d\mathbf{x},$$

where here and in what follows, $\boldsymbol{\varphi}$ denotes the predicted deformation produced by the network. By minimizing the $SSD(T, R, \boldsymbol{\varphi})$, it drives the network to learn the right parameters to ensure that $T \circ \boldsymbol{\varphi}$ looks like R as much as possible.

Ground truth difference (GTD): To guarantee that the deformation $\boldsymbol{\varphi}$ produced by the network to get close to the real labels $\boldsymbol{\varphi}_r$, we add a loss function to evaluate the sum of squared difference between these two deformations, which is defined by

$$GTD(\boldsymbol{\varphi}_r, \boldsymbol{\varphi}) = \int_{\Omega} |\boldsymbol{\varphi}_r - \boldsymbol{\varphi}|^2 d\mathbf{x}.$$

To ensure the diffeomorphism, some smoothness constraints should be added to the network. For this purpose, the regularization of the loss function is set by:

$$S(\boldsymbol{\varphi}) = \int_{\Omega} |\triangle \boldsymbol{\varphi}|^2 \, d\mathbf{x}.$$

Combing these three parts, the total loss function is formulated as follows:

$$\mathcal{L}_{loss}(\boldsymbol{\varphi}) = \lambda_1 SSD(T, R, \boldsymbol{\varphi}) + \lambda_2 GTD(\boldsymbol{\varphi}_r, \boldsymbol{\varphi}) + \lambda_3 S(\boldsymbol{\varphi}).$$

One can notice that the loss function $\mathcal{L}_{loss}(\boldsymbol{\varphi})$ includes three different hyper-parameters in the following experiment. The flowchart of the few-shot learning is referred to Fig 2.

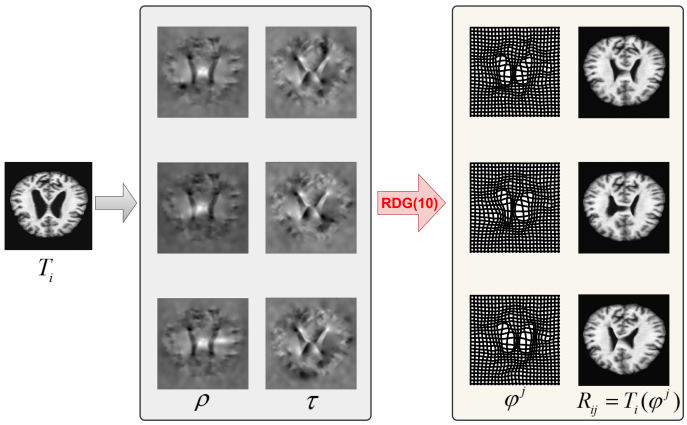


Fig. 7: The generation of diffeomorphic labels via RDG (10). First, we use the randomly number generator to generate the real component ρ and imaginary component τ (Three sets of ρ and τ are shown on the 2-3th column). Secondly, set $\tilde{\mu} = \rho + i\tau$ as input of the RDG (10) and generate the diffeomorphism $\boldsymbol{\varphi}^j$. Thirdly, we use the input image T_i to produce $R_{ij} = T_i(\boldsymbol{\varphi}^j)$.

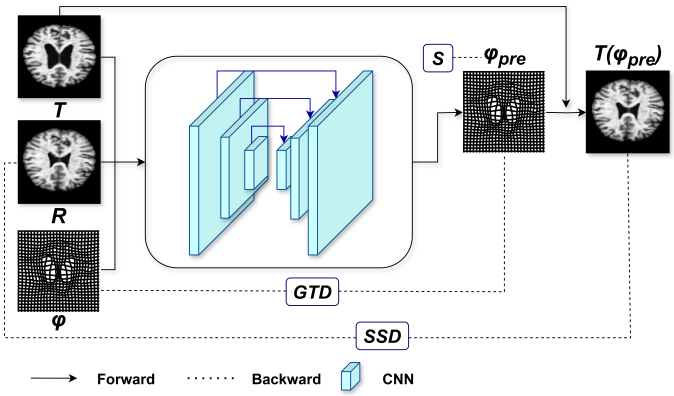


Fig. 8: The network processes input (T_i, R_{ij}) with labels $\boldsymbol{\varphi}_r^j$ to predict $\boldsymbol{\varphi}$. The loss combines: (1) $SSD(T, R, \boldsymbol{\varphi})$ for image alignment, (2) $GTD(\boldsymbol{\varphi}_r, \boldsymbol{\varphi})$ for label matching, and (3) $S(\boldsymbol{\varphi})$ for smoothness. Solid/dashed arrows denote forward/backward passes respectively.

Remark 4: The registration network's robustness could be further improved by: (i) Incorporating global anatomical guidance (e.g., organ segmentation [30]); (ii) Using randomly generated diffeomorphic labels for local deformation control. This hybrid approach could leverage both prior knowledge and learned deformation constraints.

IV. Numerical Experiments

In this section, several numerical experiments are performed to show the efficiency of the proposed network. The experiment contains five parts: In Section A and B, the datasets, implementation setup and evaluation metrics for comparison are introduced. In Section C, the comparison between supervised RDG-Unet, unsupervised RDG-Unet and Affine Generator-Uet are performed to show the advantage of the proposed supervised RDG-Unet. In Secttion D, numerical comparison between the proposed supervised RDG-Unet and other State-

of-the-Arts algorithms are performed to demonstrate the superiority of the proposed algorithms. Lastly, in Section E, we conducted few-shot learning experiment to further evaluate its performance under limited data set conditions.

A. Datasets and implementation setup

1) OASIS MRI dataset [32]: OASIS is a 3D MRI dataset, including 3D MR scans from 416 patients aged 18 to 96. For each subject, 3 or 4 individual T1-weighted MRI scans obtained in single scan sessions are included. In our numerical experiments, we evaluate the framework using test image pairs (T, R) collected from cross-patient data. Specifically: all test pairs consist of slices from different patients, and this inter-patient configuration helps evaluate generalization capability.

2) ACDC cardiac MRI dataset [33]: The ACDC is a 3D fully annotated cardiac MRI dataset containing scans from 150 patients. It includes both diastole and systole images, covering a series of left ventricle slices from the base to the apex. Each slice has labels for the myocardium and both ventricles. Fifty-four selected pairs of slices were normalized and resized to 128×128 resolution. Since both the training dataset and ACDC dataset consist of MRI scans, evaluating on this dataset demonstrates the generalization performance of the proposed method on datasets with similar structures. This dataset serves specifically for generalization testing.

3) CAMUS ultrasonic cardiogram dataset [34]: CAMUS is a 2D fully annotated ultrasonic cardiogram dataset, consists of apical four-chamber and two-chamber images from 500 patients. For each type of image, diastole and systole images with corresponding labels of the left ventricle endocardium, myocardium, and left atrium are included. We select 100 images pairs from two-chamber images as test data. To match the image size with the network input, all these images are normalized and resized to 128×128. In common sense, the registration of ultrasonic images is more difficult due to the low quality and resolution. Moreover, ultrasonic images have huge difference in structure and modality compared to the training dataset, MRI. Thus, it is a great challenge for the generalization and robustness of the proposed registration model.

4) Implementation details: The proposed network is implemented in Python with PyTorch backend on a Intel(R) Xeon(R) Platinum 8255C CPU, 40GB memory and a NVIDIA RTX 3080 GPU(10GB). The training batch size is set to be 32, and the learning rate is set to be 0.01 with an Adam optimizer. The three weights in the loss function, $\lambda_1, \lambda_2, \lambda_3$, were set to 1, 0.002, 1000 empirically.

Remark 5: As stated in the above description in this subsection, the images are down-sampled/up-sampled to the size 128×128, which makes a uniform mode for the computation of RDG (10). In fact, for the image with any size, one could also keep the size of the image unchanged and down-sampled/up-sampled the deformation $\boldsymbol{\varphi}_r^j$ (Note that $\boldsymbol{\varphi}_r^j \in C^1(\Omega)$) to the same size of the image and produce the labels $(T_i, R_{i,j}, \boldsymbol{\varphi}_r^j)$ (where $R_{i,j} = T_i(\boldsymbol{\varphi}_r^j)$) for the registration network.

B. Evaluation metrics

To show the efficiency of the proposed framework, we select the following three metrics for evaluation:

- **Relative Sum of Squared Differences (Re-SSD):** A metric to evaluate the difference between the registered image $T \circ \boldsymbol{\varphi}$ and target image R, defined by
$$\text{Re-SSD}(T, R, \boldsymbol{\varphi}) = \frac{\text{SSD}(T \circ \boldsymbol{\varphi}, R)}{\text{SSD}(T, R)},$$
where $\text{SSD}(T, R) = \sum_{i,j}(T_{i,j} - R_{i,j})^2$.
- **Dice coefficient (DC):** A metric to evaluate local alignment accuracy (i.e., specific tissues or lesion regions, which are so called regions of interest (ROIs)), defined by
$$\text{Dice}(A, B) = \frac{2|A \cap B|}{|A| + |B|},$$
where $|\cdot|$ denotes the area of the specific set, and $A = T \circ \boldsymbol{\varphi}$, $B = R$.
- **MFN:** A metric to evaluate the physcial mesh folding of the algorithm, defined by
$$\text{MFN}(\boldsymbol{\varphi}) = \#(\det(\nabla\varphi(\mathbf{x}))),$$
where $\det(\nabla\boldsymbol{\varphi}(\mathbf{x})) = (1 + \frac{\partial g_1}{\partial x_1})(1 + \frac{\partial g_2}{\partial x_2}) - \frac{\partial g_1}{\partial x_1}\frac{\partial g_2}{\partial x_2}$ and $\#\mathcal{A}$ denotes the numbers of non-positive elements in $\mathcal{A}$.

Remark 6: Concerning the above metrics, we have the following remarks:
(i). Re-SSD evaluates the difference between $T \circ \boldsymbol{\varphi}$ and target image R. The smaller Re-SSD, the better quality of the the registered results;
(ii). Dice evaluates the overlap of the specific area. Sometimes, doctors are more concerned with local alignment accuracy, such as specific tissues or lesion regions, so we use the Dice coefficient to evaluate our results when testing generalization performance with real data.
(iii). MFN evaluates the number of points in transformation field whose Jacobian determinant is less than 0 (physical mesh folding occurs). MFN=0 implies there is no mesh folding. In this view, the smaller MFN, the better quality of the transformation $\boldsymbol{\varphi}$.

C. Validation experiments

The proposed framework introduces diversity of the training data via RDG and generates diffeomorphic labels (ground truth) for the supervised network. To show the efficiency of the proposed network on controlling the physical mesh folding phenomenon, we compare the proposed RDG-UNet (supervisied) with RDG-UNet (unsupervisied) and AGUNet. Before the comparison, the three networks are trained by the following way:
RDG-UNet (supervised): Train the U-Net architecture by image pairs and corresponding transformations generated from RDG (10).
RDG-UNet (unsupervised): Train the U-Net architecture only by image pairs generated by RDG (10) without any labels on deformation $\boldsymbol{\varphi}$.

TABLE I: Performance comparison of seven registration networks on OASIS MRI dataset [32], evaluated by Re-SSD and MFN metrics (mean±std) under different training sample sizes (5/10/80/150). Set-up : we have M ($M = 5, 10, 80, 150$) raw training images T, N pairs of testing image (T, R), and W pairs of training labels $(T, R, \boldsymbol{\varphi})$, where $\boldsymbol{\varphi}$ are generated transforms; and use these N pairs to test to compare Re-SSD and MFN.

Method	Metric	5 data	10 data	80 data	150 data
VM [11]	Re-SSD	0.1699 ± 0.0207	0.1546 ± 0.0290	0.1318 ± 0.0198	0.1042 ± 0.0165
	MFN	542.52 ± 184.35	657.14 ± 214.47	464.7 ± 152.27	411.16 ± 148.46
CTF [13]	Re-SSD	0.1342 ± 0.0278	0.1307 ± 0.0239	0.0768 ± 0.0144	0.0755 ± 0.0137
	MFN	206.72 ± 83.08	227.12 ± 106.10	132.04 ± 85.71	122.12 ± 95.99
QCRegNet [12]	Re-SSD	0.1875 ± 0.0372	0.1676 ± 0.0463	0.1451 ± 0.0357	0.1358 ± 0.0448
	MFN	12.28 ± 20.48	11.96 ± 29.41	2.98 ± 8.02	$\mathbf{0.16 \pm 0.99}$
MetaRegNet [37]	Re-SSD	0.3253 ± 0.0622	0.2744 ± 0.0547	0.2315 ± 0.0493	0.2075 ± 0.0347
	MFN	7.56 ± 10.11	7.34 ± 10.26	4.72 ± 6.26	2.92 ± 5.75
AGUNet [28]	Re-SSD	0.1867 ± 0.0455	0.1254 ± 0.0244	0.1119 ± 0.0194	0.1046 ± 0.0157
	MFN	84.1 ± 61.01	92.98 ± 66.13	92.60 ± 60.87	102.24 ± 59.77
RDG-UNet (unsupervised)	Re-SSD	0.1354 ± 0.0611	0.0825 ± 0.0146	0.07925 ± 0.0161	0.0638 ± 0.0152
	MFN	55.76 ± 56.33	23.62 ± 37.55	5.2 ± 14.22	7.0 ± 10.29
RDG-UNet (supervised)	Re-SSD	$\mathbf{0.1053 \pm 0.0512}$	$\mathbf{0.0656 \pm 0.0126}$	$\mathbf{0.0476 \pm 0.0146}$	$\mathbf{0.0400 \pm 0.0085}$
	MFN	$\mathbf{1.50 \pm 5.64}$	$\mathbf{0.20 \pm 0.10}$	$\mathbf{0.16 \pm 0.70}$	0.68 ± 1.90

AGUNet: Train the U-Net architecture only by image pairs generated from Affine Generator [28], [35].

For the training data generation in numerical tests, we use the following strategy: **(1) Collecting real deformations (limited-data).** To make the generated labels more close to reality, we collect a few diffeomorphic deformations $\tilde{\boldsymbol{\varphi}}_j (j = 1, 2, \cdots, N)$ (limited-data) between real medical image pairs, and compute the corresponding Beltrami coefficients $\bar{\mu}_i (i = 1, 2, \cdots, N)$ via Eq. (4). **(2) Global and local perturbation generation.** Using the strategy in Remark 3 to generate a series of global perturbations $\mu_1^j (j = 1, 2, \cdots, M)$ and local perturbations $\mu_2^k (k = 1, 2, \cdots, P)$. This leads to the generation of randomly generated complex fields $\tilde{\mu} = \bar{\mu}_i + \mu_1^j + \mu_2^k$. **(3) Generating the diffeomorphism via RDG** (10)**.** Rescaling $\tilde{\mu}$ such that $\|\tilde{\mu}\|_\infty < 1$ and generating the diffeomorphism $\boldsymbol{\varphi}_r^j$ via Eq. (10). 150 selected images T_i are used for training and 50 selected images are used for testing for validation and comparison. The image pair for training the network is set by $(T_i, R_{ij}, \boldsymbol{\varphi}_r^j)$, where $R_{ij} = T_i(\boldsymbol{\varphi}_r^j)$. In numerical test, 4414 image labels $(T_i, R_{ij}, \boldsymbol{\varphi}_r^j)$ are used for network training. Note that the original labels come from the real image pairs between different patients, and then are randomly perturbed to generate a large amount of labels via RDG (10).

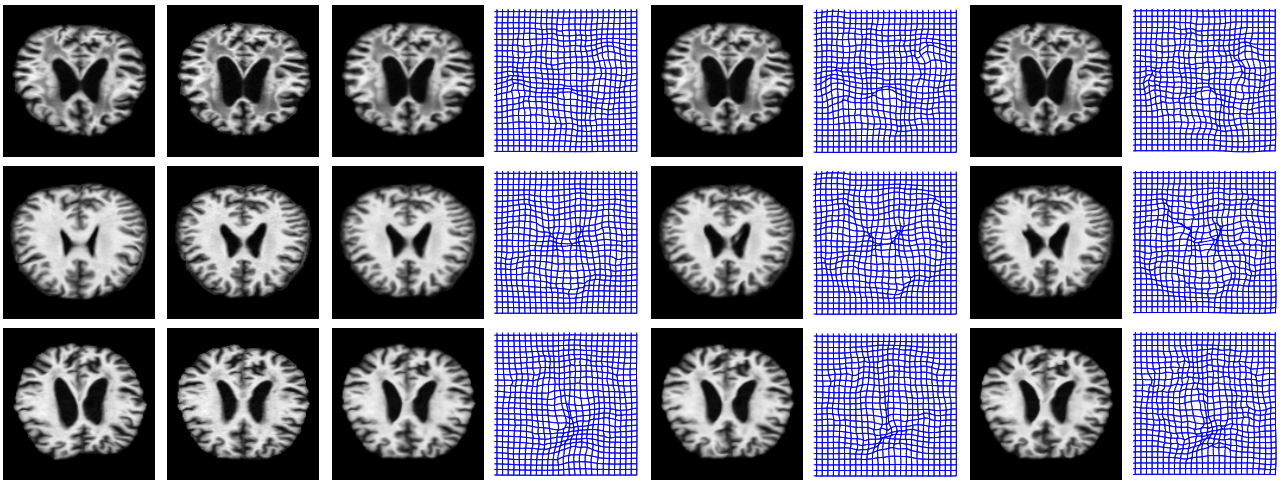
Remark 7: There are two comments on the generation of diffeomorphism:
(1). To ensure the anatomical plausibility of the randomly generated diffeomorphism, we set two positive numbers $0 < K_1 < 1 < K_2$ and drop out the generated diffeomorphisms $\boldsymbol{\varphi}$ which satisfy $\min_{\mathbf{x}\in\Omega} \det(\nabla\boldsymbol{\varphi}(\mathbf{x})) < K_1$ or $\max_{\mathbf{x}\in\Omega} \det(\nabla\boldsymbol{\varphi}(\mathbf{x})) > K_2$ (avoid excessive shrinkage or expansion); **(2).** To eliminate the effect of some generated diffeomorphims which are not representative of real applicative scenarios in network training, the real data labels are also added in the training set.

After training the above three networks by using four different levels of train data (5, 10, 80, 150) selected in OASIS MRI dataset [32], we select the other 50 images (not included in training set) as test images to compare the accuracy (Re-SSD and MFN) of these network. The quantitative comparison are listed on last six columns of Table I.

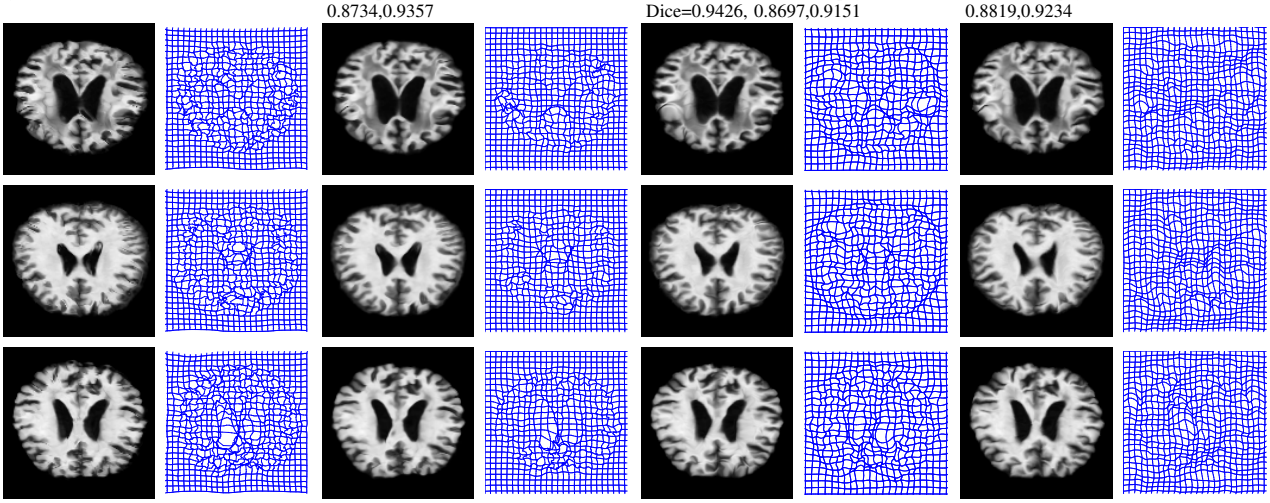
Following from the comparison between RDG-UNet (unsupervised) and RDG-UNet (supervised), one can notice that the RDG-UNet (supervised) achieves better registration accuracy on Re-SSD (with value of 0.0400 ± 0.0085) and nearly 0 value of MFN at all levels of training data set. The superiority for controlling the mesh folding is especially obvious when the amount of training data is small. This shows the advantage of supervised RDG-UNet, and it is also the main motivation for us to propose the RDG for supervised few-shot learning. Besides, in some view, this comparison can also be viewed as an ablation experiment which shows the necessity to introduce the RDG to generate diffeomorphic labels for supervised learning network.

In another view, the comparison between AGUNet and RDG-UNet (supervised) shows that the finite dimensional space (the solution of AGUNet [28], [35] is determined by six parameters) is not enough to simulate the deformable deformations. This is another motivation for us to introduce RDG to generate infinite dimensional diffeomorphisms.

In addition, to show the proposed model's ability to consistently perform over multiple slices, we select the 3D images of three patients in OASIS dataset [32] and use the proposed

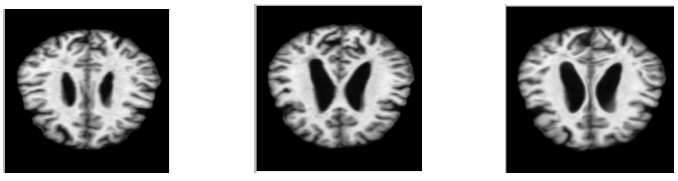


(a) T (registered image) (b) R (target image) (c) Ours: The registration result and the physical mesh. Dice=0.9321, 0.8734,0.9357 (d) Ours (unsupervised): The registration result and the physical mesh. Dice=0.9426, 0.8697,0.9151 (e) AGUNet [28]: The registration result and the physical mesh. Dice=0.9374, 0.8819,0.9234



(f) VM [11]: The registration result and the physical mesh. Dice=0.9293, 0.8684,0.9273 (g) CTF [13]: The registration result and the physical mesh. Dice=0.9292, 0.8288,0.8652 (h) QCRegNet [12]: The registration result and the physical mesh. Dice=0.9321, 0.8821,0.9060 (i) MetaReg [37]: The registration result and the physical mesh. Dice=0.9284, 0.7949,0.8573

Fig. 9: Few-shot tests based on 5 original data. The registration result of our method and other state-of-art which are all trained on 5 original data. (a) and (b) are T and R of three different image pairs. (c)-(h) are the registration results of 7 different networks. The places where the mesh of deformation intersects represent mesh folding.



T R $T(\varphi)$

Fig. 10: Registration result of the proposed few-shot learning network between 2D slices from the 3D data of one patient, where T and R are floating image and target image, $T(\varphi)$ is the registration result and φ is the output deformation of the proposed network.

network to perform registration between the 2D slices pairs, it shows a satisfactory performance for the proposed network with Re-SSD=0.2369 ± 0.0266, MFN=10.20 ± 7.33 (One registration result is listed on Fig 10).

Based on the above comparison, we conclude that the proposed framework, which brings more various and stronger physical information to the network, can greatly enhance the capability of the network. This provides evidence to show the efficiency of the proposed few-shot learning framework for supervised diffeomorphic image registration.

D. Comparison with State-of-Arts registration algorithms

Followed the comparison in Subsection C, in this subsection, we compare the proposed few-shot learning network with other State-of-Arts registration algorithms including:

SyN: a classical traditional method, using the SyNOnly setting in ANTS [36].

VoxelMorph (VM) : a popular single-stage registration network [11].

CTF: a coarse to fine unpervised registration network com-

posed of a affine transformation and deformable transformation [13].

QCRegNet (QC): a diffemorphic registration network based on quasi-conformal theory [12].

MetaRegNet (MR): a deep metamorphic image registration network [37], who has the ability to address few-shot learning for image registration.

The quantitative comparison results are shown on Table I and Table II (Comparison on OASIS MRI dataset [32] is listed on Table I, and real data comparison on ACDC cardiac MRI dataset [33], CAMUS ultrasonic cardiogram dataset [34] are listed on Table II), and some slice comparison results of the test image are also listed on Fig 9.

As it is shown on Table I and Table II, the proposed framework achieves best performance on Re-SSD and MFN. It is worth to notice that even if the results of QCRegNet [12] on Re-SSD falls short of expectations, the performance on MFN is comparable to our model. This is because of the fact that QCRegNet [12] also introduces a mechanism to learn diffeomorphism, though it may rely on large data set for training the network. Concerning the other two networks, there is still a big gap for VM and CTF in avoiding physical mesh folding (MFN is much bigger than the proposed network). In addition, from Fig 9, one can notice that the deformation produced by the proposed framework is of high smoothness while the deformation in QCRegNet [12] is lack of smoothness. This is because of the fact that the proposed framework minimizes $\lambda\bar{\mathcal{F}}(\boldsymbol{\varphi})+\alpha\mathcal{G}(\boldsymbol{\varphi})$ and the solution is regularized by $\mathcal{G}(\boldsymbol{\varphi})$, while the network in QCRegNet [12] only minimizes the fitting term $\mathcal{F}(\boldsymbol{\varphi})$ without any regularization. This is the main motivation to add the Cauchy-Riemann constraint $\mathcal{G}(\boldsymbol{\varphi})$ in RDG model (10).

E. Few-shot learning test and generalization test

In this subsection, we reduce the number of image in training data sets to simulate few-shot learning scenarios where the training data is not enough. For this purpose, we conducted numerical tests by limiting the training dataset size to 5,10, 80 and 150 samples, respectively. After training the corresponding networks, different test image pairs are set as input of these networks to get the Re-SSD and MFN. The comparison results for different levels of training data are listed on Table I and Fig 12. From Table I and Fig 12, one can notice that the proposed framework has the ability to achieve a very accurate nearly without mesh folding under conditions of limited data availability. This shows the good performance of the proposed few-shot learning framework. For the dataset with 5 original images, we manually delineated the labels of the central cavity of the brain and calculated the corresponding Dice metrics after registration (see Fig 9). This shows that the proposed framework also has good performance on the aspect on Dice metric.

Remark 8: We also tested the registration efficiency in the extreme case (with only one original image), and the experimental results on the same test set reached Re-SSD$=0.1574\pm0.0391$, MFN$=5.22\pm0.90$. An example is listed in Fig 11.

Fig. 11: One registration result of the test set for the proposed few-shot learning network with only one original image in training set, where T and R are floating image and target image, $T(\varphi)$ is the registration result and φ is the output deformation of the proposed network

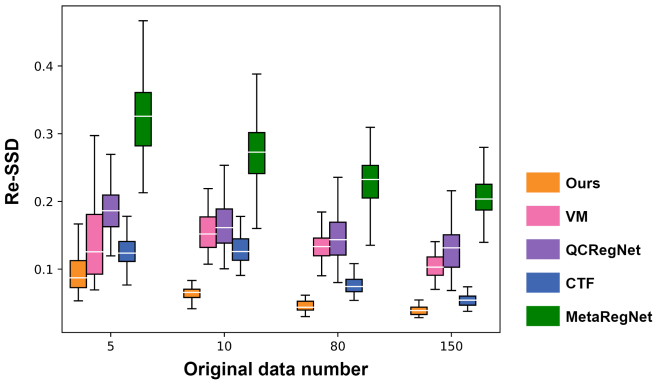
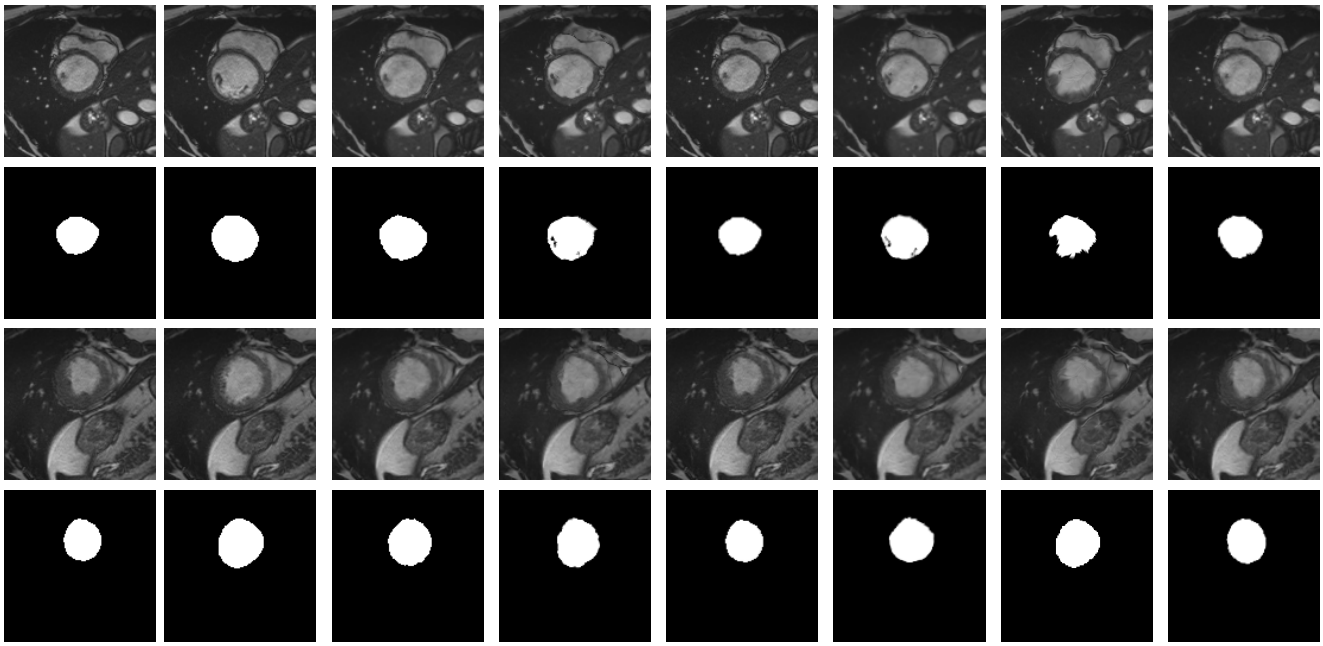


Fig. 12: Boxplot (Re-SSD) of 5 different registration algorithms on different levels $(5, 10, 80, 150)$ of original training sets, where the vertical ordinate denotes the Re-SSD of registration of 5 different algorithms on different levels. Note that here the average self structural similarity (SSIM) of the training set of for 5-original image is 0.65, and the SSIM between 5-original training set and 150-original training set is 0.52.

In addition, to show the robustness of the proposed framework, we perform the generalization tests on ACDC cardiac MRI dataset [33] and CAMUS ultrasonic cardiogram dataset [34]. In this two datasets, segmented labels of ROIs are provided for each image. To assess the performance, we use the proposed few-shot learning network to register each image pair. Another five deep learning networks are also performed on these two datasets. Based on the registration results of these six networks, the Dice between the labels of each registered image pair $(T\circ\boldsymbol{\varphi}, R)$ is reported. After training six different networks with 150 samples, we use the 2D slices in ACDC cardiac MRI dataset [33] and CAMUS ultrasonic cardiogram dataset [34] to test the performance of these six networks. The comparisons are listed on Table II and Fig 13-14. From the comparison, we conclude that the proposed few-shot learning framework for supervised diffeomorphic image registration network has the ability to achieve accurate registration with a few training data, and has the ability to adapt to different kinds of dataset.

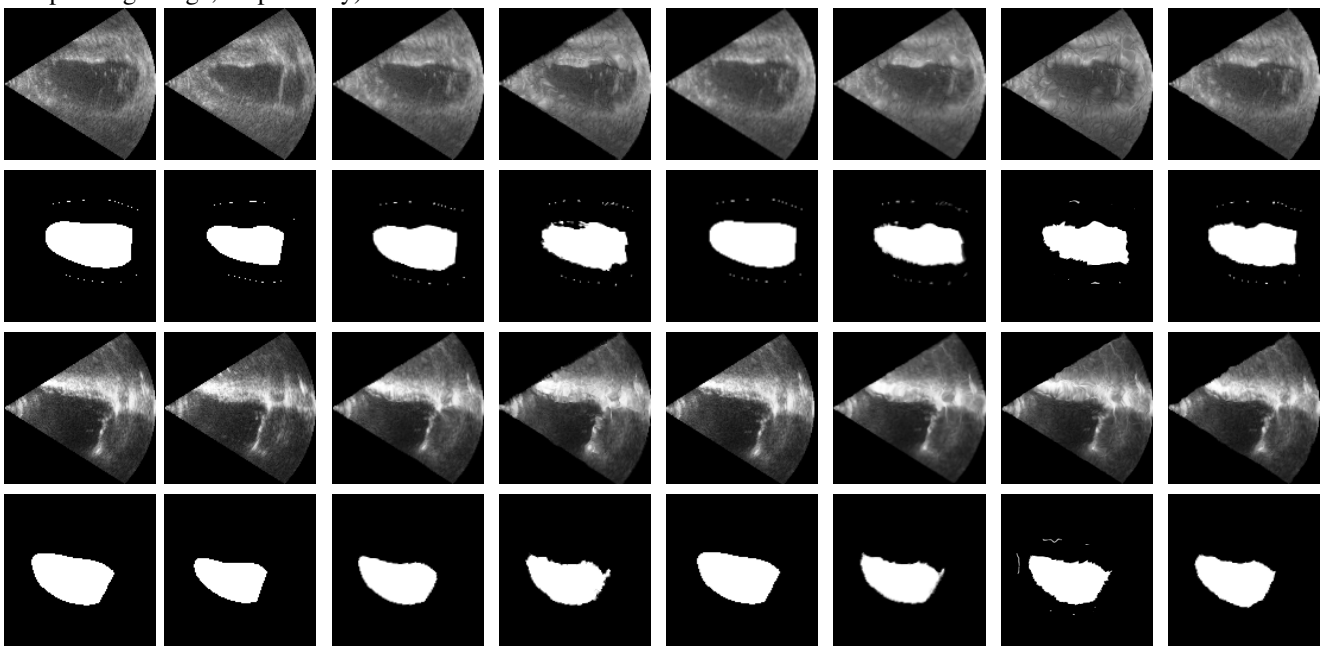
V. Sensitivity Test for Hyper-Parameters

In previous section, we empirically set the loss function parameters $(\lambda_1, \lambda_2, \lambda_3)$ to be $(1, 0.002, 1000)$. This section



(a) T (registered image) (b) R (target image) (c) Ours (d) VM [11] (e) SyN [36] (f) CTF [13] (g) QC [12] (h) MR [37]

Fig. 13: Generalization tests on ACDC dataset [33]: The first column presents two different registered images T; The second column presents two different target images R; The 3rd-8th column presents the registration results $T \circ \varphi$ of the proposed network, VM [11], SyN [36], CTF [13], QC [12] and MR [37] (The second row and fourth row denote the label of ROI of corresponding image, respectively).



(a) T (registered image) (b) R (target image) (c) Ours (d) VM [11] (e) SyN [36] (f) CTF [13] (g) QC [12] (h) MR [37]

Fig. 14: Generalization tests on CAMUS dataset [34]: The first column presents two different registered images T; The second column presents two different target images R; The 3rd-8th column presents the registration results $T \circ \varphi$ of the proposed network, VM [11], SyN [36], CTF [13],QC [12] and MR [37] (The second row and fourth row denote the label of ROI of corresponding image, respectively).

TABLE II: Comparison between the proposed framework and some State-of-Arts registration algorithms on ACDC cardiac MRI dataset [33] (2D slices are selected from the 3D dataset) and CAMUS ultrasonic cardiogram dataset [34] (2D dataset). All methods use our new set: train on W pairs (or tuples (T, R, φ)) and test on N pairs (W and N are referred to Table I).

	Dice	
	ACDC [33]	**CAMUS [34]**
SyN [36]	0.6893 ± 0.1703	0.6399 ± 0.1011
VM [11]	0.7384 ± 0.1654	0.7320 ± 0.0845
CTF [13]	0.7398 ± 0.1193	0.6672 ± 0.0619
QC [12]	$\mathbf{0.7971 \pm 0.1442}$	0.6795 ± 0.1000
MR [37]	0.7485 ± 0.1650	0.7625 ± 0.0794
Ours	0.7960 ± 0.1366	$\mathbf{0.7877 \pm 0.0678}$

presents a systematic parameter sensitivity analysis where we have fixed λ_1 and varying λ_2 (0 to 0.1 with increments of 0.01), varying λ_3 (0 to 2000 with increments of 200). For each parameter configuration, we evaluate their impacts on registration performance through Re-SSD and MFN metrics. Fig 15 reveals three key observations: (i) increasing λ_3 leads to a smaller MFN value and larger Re-SSD, indicating that $S(\varphi)$ effectively enhances deformation field smoothness. However, excessive weight may lead to the deformation to be over-smoothed, consequently impairing registration accuracy; (ii) the introduction of the GTD (when $\lambda_2 > 0$) substantially reduces MFN without significantly affecting Re-SSD, demonstrating that the RDG-generated labels can effectively suppress mesh folding while maintaining registration precision; (iii) the sensitivity test for hyper-parameters shows that our framework maintains consistent robustness across parameter variations and achieves effective few-shot learning performance.

To further validate the robustness of the proposed framework, we conducted cross-validation experiments using a dataset of 150 samples. The data was systematically partitioned into different training and test sets through random stratified sampling. After training the network with randomly selected training set, we applied the proposed network to perform registration on the test sets. The experimental results demonstrated consistent performance across various data partitions, achieving a registration accuracy of Re-SSD = 0.0499 ± 0.0115, MFN = 0.12 ± 0.84 and Re-SSD = 0.0825 ± 0.0184, MFN = 0.86 ± 3.38. This result is align with the quantitative results shown in Table 9, further validating the robustness and generalization capability of the proposed network under different data division scenarios.

VI. Conclusion

Mesh folding elimination is important for many image registration applications in clinical medicine. Existing registration methods are limited either by the lack of labels for supervised learning or the lack of physical prior for unsupervised learning. In this paper, we propose a novel few-shot learning registration framework for supervised diffeomorphic image registration

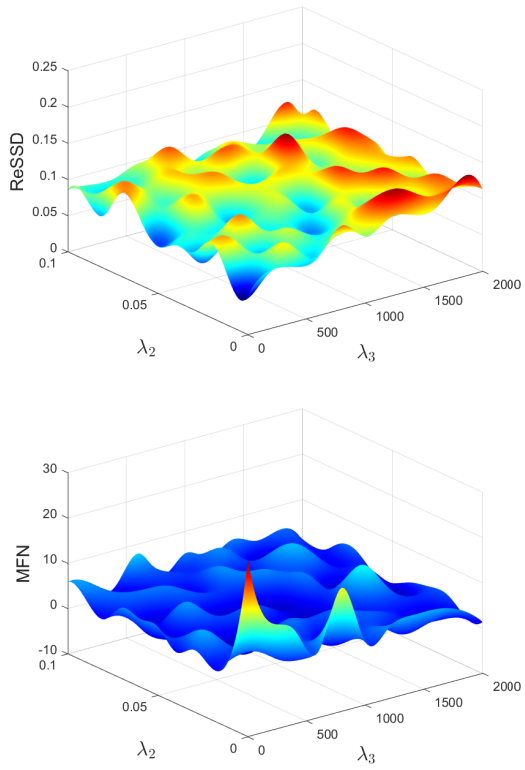


Fig. 15: Influence of the weights (λ_2 and λ_3) in the loss function on the registration performance (Re-SSD and MFN).

network. The proposed framework contains three parts: the random diffeomorphism generator (RDG) which inverts the real labels φ_r via the information provided by the randomly generated complex vector fields, network training data generator which generates the labels for training the supervised network, and supervised learning network for diffeomorphic image registration.

The proposed framework is not limited to large amount of dataset for training and has the ability to be trained well under the condition that only a few images can be used for training. This is very important for medical image registration on some rare diseases in which case the collection of images is challenging.

Though the proposed few-shot learning framework has many advantages, limitations still exists on the following three aspects: **(1).** The RDG and the registration network are independent and the no end-to-end mode needs manual intervention; **(2).** The quasiconformal theory works only on 2D space, which makes it difficult to extend to 3D registration directly; **(3).** The original labels rely on the real data. To address the above three challenges, in future research, we main focus on the following three problems: **(i).** Extending the framework to a model composed entirely of neural networks which ensures an end-to-end mode for few-shot learning; **(ii).** Extending the proposed 2D few-shot learning framework to 3D few-shot learning framework (The rough idea is referred to Appendix for details. Though the principle seems to be direct, there are still some difficulties for this extension. We

will address these challenges in forthcoming work); **(iii).** Replacing the U-Net structure in the proposed framework with some more complex networks (for example, the Generative Adversarial Networks (GAN)) to address the dependence on the real data.

APPENDIX:3D EXTENSION OF THE PROPOSED FEW-SHOT LEARNING FRAMEWORK

Assume Ω is a bounded domain on $\mathbb{R}^3$ and $\boldsymbol{\varphi}:\Omega\to\Omega$ is a diffeomorphism, then for any point $\mathbf{x}\in\Omega$, by polar decomposition, we have

$$\begin{aligned}\nabla\boldsymbol{\varphi}(\mathbf{x}) =&U(\mathbf{x})P(\mathbf{x})=U(\mathbf{x})\sqrt{\nabla^T\boldsymbol{\varphi}(\mathbf{x})\nabla\boldsymbol{\varphi}(\mathbf{x})}\\ =&U(\mathbf{x})W(\mathbf{x})\Sigma(\mathbf{x})W^T(\mathbf{x}),\end{aligned}\tag{24}$$

where $U(\mathbf{x})$ and $W(\mathbf{x})$ are rotation matrices, $\Sigma(\mathbf{x})$ is the diagonal matrix containing the singular values a, b and c of $\nabla\boldsymbol{\varphi}(\mathbf{x})$. By convention, we require that $a\geq b\geq c>0$.

Note that here $W(\mathbf{x})$ is a 3D rotation matrix, which is denoted by

$$\begin{aligned}W(\mathbf{x}) =&W_{x_3}(\mathbf{x})W_{x_2}(\mathbf{x})W_{x_1}(\mathbf{x}),\\ =&\begin{pmatrix}r_{11}&r_{12}&r_{13}\\ r_{21}&r_{22}&r_{32}\\ r_{31}&r_{32}&r_{33}\end{pmatrix},\end{aligned}\tag{25}$$

where $W_{x_1}(\mathbf{x})=\begin{pmatrix}1&0&0\\ 0&\cos\theta_x&-\sin\theta_x\\ 0&\sin\theta_x&\cos\theta_x\end{pmatrix}$,

$W_{x_2}(\mathbf{x})=\begin{pmatrix}\cos\theta_y&0&-\sin\theta_y\\ 0&1&0\\ \sin\theta_y&0&\cos\theta_y\end{pmatrix}$,

$W_{x_3}(\mathbf{x})=\begin{pmatrix}\cos\theta_z&-\sin\theta_z&0\\ \sin\theta_z&\cos\theta_z&0\\ 0&0&1\end{pmatrix}$, $r_{11}=\cos\theta_z\cos\theta_y$,
$r_{12}=\cos\theta_z\sin\theta_y\sin\theta_x-\sin\theta_z\cos\theta_x$,
$r_{13}=\cos\theta_z\sin\theta_y\cos\theta_x-\sin\theta_z\sin\theta_x$, $r_{21}=\sin\theta_z\cos\theta_y$,
$r_{22}=\sin\theta_z\sin\theta_y\sin\theta_x+\cos\theta_z\cos\theta_x$,
$r_{23}=\sin\theta_z\sin\theta_y\cos\theta_x-\cos\theta_z\sin\theta_x$, $r_{31}=-\sin\theta_y$, $r_{32}=\cos\theta_y\sin\theta_x$, $r_{33}=\cos\theta_y\cos\theta_x$.

The angles $\theta_x=\arctan(r_{32}/r_{33})$,
$\theta_y=\arctan(-r_{31}/\sqrt{r_{32}^2+r_{33}^2})$, $\theta_z=\arctan(r_{21}/r_{11})$.

By Eq. (25), we obtain the 3D quasiconformal representation [38] for the mapping $\boldsymbol{\varphi}:\Omega\to\Omega$ by

$$\mu(\mathbf{x})=(a,b,c,\theta_x,\theta_y,\theta_z)\qquad\forall\mathbf{x}\in\Omega.\tag{26}$$

Based on this representation, the 2D model (6) is directly extended to 3D model

$$\boldsymbol{\varphi}=\arg\min\lambda\mathcal{F}(\boldsymbol{\varphi})+\alpha\mathcal{G}(\boldsymbol{\varphi})\tag{27}$$

where $\tilde{\mu}(\mathbf{x})$ is a given six dimensional vector field with the first three components are positive, $\lambda,\alpha>0$, the fitting term

$$\mathcal{F}(\boldsymbol{\varphi})=\int_\Omega|\tilde{\mu}-\mu(\boldsymbol{\varphi})|^2d\mathbf{x},\tag{28}$$

with $\mu(\boldsymbol{\varphi})$ is a 3D quasiconformal representation for the mapping $\boldsymbol{\varphi}:\Omega\to\Omega$ determined by Eq. (26) and the 3D conformal regularization [39]

$$\mathcal{G}(\boldsymbol{\varphi})=\int_\Omega\|\nabla\boldsymbol{\varphi}\nabla^T\boldsymbol{\varphi}-\frac{\|\nabla\boldsymbol{\varphi}\|_F^2}{3}\mathbf{I}\|^2d\mathbf{x}.\tag{29}$$

In Eq. (27), it is not explicit for the definition of $\mu(\boldsymbol{\varphi})$, though it may be suitable for being set as a loss function for some back propagation network. To make it explicit, we further discuss the reconstruction of 3D quaisconformal mapping $\boldsymbol{\varphi}:\Omega\to\Omega$ by giving a six dimensional vector field $\tilde{\mu}(\mathbf{x})=(\tilde{a},\tilde{b},\tilde{c},\tilde{\theta}_x,\tilde{\theta}_y,\tilde{\theta}_z)$ with $\tilde{a},\tilde{b},\tilde{c}>0$.

By Eq. (24), there holds,

$$\nabla^T\boldsymbol{\varphi}\nabla\boldsymbol{\varphi}=W\begin{pmatrix}a^2&0&0\\ 0&b^2&0\\ 0&0&c^2\end{pmatrix}W^T.\tag{30}$$

Right multiplying by $(\nabla\boldsymbol{\varphi})^{-1}$ on both side of Eq. (30), we have

$$\nabla^T\boldsymbol{\varphi}=W\begin{pmatrix}a^2&0&0\\ 0&b^2&0\\ 0&0&c^2\end{pmatrix}W^T\frac{1}{\det(\nabla\boldsymbol{\varphi})}C,\tag{31}$$

where C is the adjugate matrix of $\nabla\boldsymbol{\varphi}$ with $C=(\nabla\varphi_2\times\nabla\varphi_3,\nabla\varphi_3\times\nabla\varphi_1,\nabla\varphi_1\times\nabla\varphi_2)$.

Define $A=W\begin{pmatrix}\frac{bc}{a}&0&0\\ 0&\frac{ac}{b}&0\\ 0&0&\frac{ab}{c}\end{pmatrix}W^T$, Eq. (31) becomes

$$A\nabla^T\boldsymbol{\varphi}=C,\tag{32}$$

Note that here A is determined by the giving the vector field $\tilde{\mu}(\mathbf{x})=(\tilde{a},\tilde{b},\tilde{c},\tilde{\theta}_x,\tilde{\theta}_y,\tilde{\theta}_z)$.

Based on Eq. (32), we proposed the other variational model to generate 3D quaisconformal mapping:

$$\boldsymbol{\varphi}=\arg\min\lambda\bar{\mathcal{F}}(\boldsymbol{\varphi})+\alpha\mathcal{G}(\boldsymbol{\varphi}),\tag{33}$$

where $\tilde{\mu}(\mathbf{x})$ is a given six dimensional vector field with the first three components are positive, $\lambda,\alpha>0$, the fitting term

$$\mathcal{F}(\boldsymbol{\varphi})=\int_\Omega\|A\nabla^T\boldsymbol{\varphi}-C\|^2d\mathbf{x}.\tag{34}$$

ACKNOWLEDGMENT

This work was supported by the National Key Research and Development Program of China (2020Y-FA0714200), the National Natural Science Foundation of China (No. 12171379, 12271417 and 12471484) and the Fundamental Research Funds for the Central Universities, China. Thanks to the referees for their very helpful remarks.

REFERENCES

[1] J.P. Thirion, Image matching as a diffusion process: an analogy with Maxwell's demons, *Medical Image Analysis*, 2, 1998:243–260.

[2] N. Chumchob, Vectorial total variation-based regularization for variational image registration, *IEEE Transactions on Image Processing*, 22, 2013: 4551-4559.

[3] Z. Nie and X. Yang, Deformable image registration using functions of bounded deformation, *IEEE Transactions on Medical Imaging*, 38 (6), 2019: 1488-1500.

[4] Q. Zeng and Y. Chen.Accurate Inverse Consistent Non-rigid Image Registration and Its Application on Automatic Re-contouring. *Bioinformatics Research and Applications*,4983, 2008: 293-304.

[5] L. K. Chun and L. M. Lui, Landmark and intensity based registration with large deformations via quasiconformal maps, *SIAM Journal on Imaging Sciences*, 7(4), 2014:2364-2392.

[6] H. Han and Z. Wang, A diffeomorphic image registration model with fractional-order regularization and Cauchy-Riemann constraint, *SIAM Journal on Imaging Sciences*, 13(3), 2020: 1240-1271.

[7] Y. T. Lee, K. C. Lam, and L. M. Lui, Landmark-matching transformation with large deformation via n-dimensional quasi-conformal maps, *Journal of Scientific Computing*, 67,2016: 926–954.

[8] J. Zhang, K. Chen, and B. Yu, *A novel high-order functional based image registration model with inequality constraint.* Computers and Mathematics with Applications, 72(12), 2016: 2887–2899.

[9] N. Debroux, C. Le Guyader, and L. Vese, A multiscale deformation representation, *SIAM Journal on Imaging Sciences*, 16 (2), 2023:802–841.

[10] P. Dupuis, U. Grenander, and M. I. Miller, Variational problems on flows of diffeomorphisms for image matching, *Quarterly of Applied Mathematics*, 56(3), 1998:587–600.

[11] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag, and A. V. Dalca, Voxelmorph: a learning framework for deformable medical image registration, *IEEE Transactions on Medical Imaging*, 38(8), 2019: 1788-1800.

[12] Q. Chen, Z. Li, L. M. Lui. A deep learning framework for diffeomorphic mapping problems via quasi-conformal geometry applied to imaging, *SIAM Journal on Imaging Sciences*, 17(1), 2024: 501-539.

[13] W. Huang, H. Yang, X. Liu, C. Li, I. Zhang, R. Wang, H. Zheng, and S. Wang, A coarse-to-fine deformable transformation framework for unsupervised multi-contrast mr image registration with dual consistency constraint, *IEEE Transactions on Medical Imaging*, 40(10), 2021: 2589.

[14] M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu, Spatial transformer networks, *MIT Press, 2015*.

[15] X. Jia, A. Thorley, W. Chen, H. Qiu, et al., Learning a model-driven variational network for deformable image registration, *IEEE Transactions on Medical Imaging*, 41(1), 2022: 199-212.

[16] L C. Evans, *Partial differential equations.* Providence, Rhode Island, 2010.

[17] W. Zeng, J. Marino, K. C. Gurijala, X. Gu, and A. Kaufman, Supine and prone colon registration using quasi-conformal mapping, *IEEE Transactions on Visualization and Computer Graphics*, 16, 2010:1348–1357.

[18] L. M. Lui et al., Optimization of surface registrations using beltrami holomorphic flow, *SIAM Journal on Scientific Computing*, 50(3), 2012: 557–585.

[19] H. Han and A. Wang, A fast multi grid algorithm for 2D diffeomorphic image registration model, *Journal of Computational and Applied Mathematics*, 394 (2021): 113567.

[20] H. Han, Z. Wang, and Y. Zhang, MultiScale approach for Two-Dimensional diffeomorphic image registration, *SIAM Multiscale Modeling and Simululation*, 19(4), 2021:1538–1572.

[21] H. Han and Z. Wang, 3D diffeomorphic image registration with Cauchy-Riemann constraint and lower bounded deformation divergence, *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(1) , 2023: 299-328.

[22] Bob D. de Vos, Floris F. Berendsen, Max A. Viergever, Hessam Sokooti, Marius Staring, Ivana Isgum, A deep learning framework for unsupervised affine and deformable image registration, *Medical Image Analysis*, 52, 2019: 128–143.

[23] X. Jia, W. Lu, X. Cheng and J. Duan, Decoder-Only Image Registration, *IEEE Transactions on Medical Imaging*, Online.

[24] H.Jonathan, J. Ajay, A. Pieter, Denoising diffusion probabilistic models, *Advances in Neural Information Processing Systems*, 2020:33.

[25] Y. Zhuo and Y. Shen, DiffuseReg: denoising diffusion model for obtaining deformation fields in unsupervised deformable image registration, *International Conference on Medical Image Computing and Computer-Assisted (MICCAI)*, 27 (2024):597–607.

[26] Y. Qin, X. Li, FSDiffReg: Feature-wise and Score-wise Diffusion-guided Unsupervised Deformable Image Registration for Cardiac Images, *International Conference on Medical Image Computing and Computer-Assisted (MICCAI), Lecture Notes in Computer Science*, 14229 (2023): 655–665.

[27] S. Ying, W. Yang, S. Du, J. Shi, Deep learning based medical image registration: a review, *Pattern Recognition and Artificial Intelligence*, 34(4), 2021: 287-299.

[28] E. Chee, Z. Wu, AIRNet: Self-Supervised Affine Registration for 3D Medical Images using Neural Networks, *arXiv*, 2018.

[29] J. Chen, E. Frey and Y. He et al., TransMorph: Transformer for unsupervised medical image registration, *Medical Image Analysis*, 82 (2022):102615.

[30] M. Hoffmann, B. Billot and D. Greve et al., SynthMorph: learning contrast-invariant registration without acquired images, *IEEE Transactions on Medical Imaging*, 41(3) (2021):543–558.

[31] N. Chumchob and K. Chen, A robust multigrid approach for variational image registration models, *Journal of Computational and Applied Mathematics*, 236(5), 2011: 653-674.

[32] Marcus, DS, Wang, TH, Parker, J, Csernansky, JG, Morris, JC, Buckner, RL. Open Access Series of Imaging Studies (OASIS): Cross-Sectional MRI Data in Young, Middle Aged, Nondemented, and Demented Older Adults. *Journal of Cognitive Neuroscience*, 19, 2007:1498-1507.

[33] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, et al., Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Transactions on Medical Imaging,* 37 (11), 2018: 2514–2525.

[34] S. Leclerc, E. Smistad, J. Pedrosa, A. Ostvik, et al., Deep learning for segmentation using an open largescale dataset in 2D echocardiography, *IEEE Transactions on Medical Imaging*, 38(9) ,2019:2198–2210.

[35] P. Dong and N. P. Galatsanos, Affine transformation resistant watermarking based on image normalization, *International Conference on Image Processing* 3, 2002: 489–492.

[36] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee, Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain, *Medical image analysis*, 12(1), 2008:26–41.

[37] J. Ankita,Y. Hong, MetaRegNet: Metamorphic image registration using flow-driven residual networks[J]. *Lecture Notes in Computer Science*,(14243),2023: 160-170.

[38] Q. Chen, and L. M. Lui, A novel 3D mapping representation and its applications, *arXiv:2308.05333*, (2023).

[39] H. Han, Z. Wang and Y. Zhang Multiscale approach for Three-Dimensional conformal image registration, *SIAM Journal on Imaging Sciences*, 15(3) (2022):1413-1468.