HELMHOLTZ'S THEOREM BASED PARAMETRIC NON-RIGID IMAGE REGISTRATION

by

HSI-YUE S. HSIAO

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON
December 2008

In memory of my grandparents, Fang-Pu and Chi Hsiu-Ying Pai

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to every person who helped me throughout my doctoral work. I gratefully acknowledge my supervising professor, Dr. Hua-Mei Chen, for his invaluable supervision, guidances, and advices. He was always there to listen and lead me back to the correct tracks.

I would also like to seize this opportunity to extend my sincere gratitude to my committee members Dr. Jean Gao, Dr. Guojun Liao, and Dr. Roger Walker who provided inspirations, supports, time commitment and indispensable expertise.

I am grateful to my teammates Joseph Hsieh and Ike Lin who shared the ideas and findings disinterestedly from time to time.

I would also like to give my profound appreciation to my beloved grandparents and parents for their love and care from childhood. Thanks also go to my sister for her selfless support.

Finally, I would like to express my special appreciation to my loving wife, Jane, for all of her support, encouragement and love.

November 21, 2008

ABSTRACT

HELMHOLTZ'S THEOREM BASED
PARAMETRIC NON-RIGID
IMAGE REGISTRATION

HSI-YUE S. HSIAO, Ph.D.

The University of Texas at Arlington, 2008

Supervising Professor: Hua-Mei Chen

Helmholtzs theorem states that, with suitable boundary condition, a vector field is completely determined if both of its divergence and curl are specified everywhere. Based on this, we developed a new parametric non-rigid image registration algorithm. Instead of the displacements of regular control grid points, the curl and divergence at each grid point are employed as the parameters. This leads to a very simple mathematical model - two Poisson equations in 2-D or three Poisson equations in 3-D which will be used to solve for the displacement field. The similarity measure used in the method is sum of squared difference. And multi-resolution is applied on the gradient descent optimization.

In this dissertation, two closest related works are reviewed. The first one is the fast parametric elastic image registration which the parameters are the B-spline coefficients of the displacement field at each control grid point. However, in the fast parametric elastic image registration method, it is very likely to result in grid folding in the final deformation field if the distance between adjacent control grid points

(knot spacing) is less than 8. This implies that the high frequency components in the deformation field can not be accurately estimated. Another relevant work is the parametric non-rigid image registration method based on Helmholtz decomposition. In that work, the deformation is considered as the result of two types of particles, namely the vortex particles and sink and source particles. Three types of parameters are associated with each particle: 1. the vorticity (curl) or divergence strength, 2. the influence domain, 3. position. Their method leads to a very complicated mathematic model and cannot be generalized to 3-D case easily. On the contrary, in our case, the divergence and curl, used as the only control parameters, are associated with each grid point. This leads to a very simple mathematical model and can be applied to 3D case easily.

Though the present work does not guarantee the regularity (no mesh folding) of the resulting deformation field, it is rarely occurred in the practical case. However, to keep the completeness of the proposed method, we set the parameters as the divergence and curl of the B-spline coefficients of displacement field which further strengthen the regularization to be strong enough to warrant a folding free deformation field.

In the experiments, we illustrate that the proposed method has the capability to handle various real medical images for both 2-D and 3-D. Also, an experiment was devised to show the proposed method can overcome the aperture problem.

TABLE OF CONTENTS

Chapter

LIST OF FIGURES

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

Medical imaging technologies have been broadly used in many medical professions nowadays. Many radiology equipments were invented in the past 50 years such as Ultra Sound (US) [1], Magnetic Resonance (MR) [1], Computed Tomography (CT) [1], and Positron Emission Tomography (PET) [1]. These devices can produce valuable images that help doctors to diagnose diseases. However, images taken from different time or devices are hard to be compared. Image registration is a solution to provide useful support for this purpose.

Image registration is a fundamental task in image processing. It transforms different sets of data into one coordinate system. Registration is necessary in order to compare or integrate data obtained from different imaging modalities. It has great applications in areas like remote sensing [2] (cartography updating), medical imaging [3] (change detection, tumor monitoring), and computer vision [4, 5].

Image registration problem can be classified into two categories - rigid and non-rigid [6]. In rigid image registration, the transformation between an image pair can be described by a small set of global variables such as translations, rotations and scalings. In non-rigid image registration, the transformation between an image pair cannot be obtained in terms of global translations, rotations, and scalings. It needs some localized image stretching to achieve registration. This work focuses on the latter case.

A basic non-rigid image registration method consists of a transformation model, which is applied to reference image coordinates to locate their corresponding coordinates in the template image space, an optimization method, which maximizes the similarity of image pairs, and a similarity measure which measures the similarity between an image pair subject to a given transformation. The general non-rigid image registration problem can be defined as following. Given a reference image R and a template image T defined on a domain Ω, we want to find a transformation from Ω to Ω which makes R and transformed T has the best overlap. The transformation is represented by $\varphi(x) = x - \phi(x)$ where ϕ is the displacement field such that $\phi : \Omega \to \Omega$. The task is how to find a mapping $\varphi = (\varphi_1, ..., \varphi_n)$ to transform the template image $T_\varphi = T(\varphi(x))$ such that T_φ is similar to R in the geometrical sense. To find the mapping φ, one must optimize the similarity measure and at the same time, maintain the regularity of the transformation. Here the regularity means that the transformation has to be physically reasonable, or mathematically, a injective function. In image registration, sum or the squared differences (SSD) and mutual information (MI) [7] are commonly used as similarity measure. Fig. 1.1 illustrates the general idea of a non-rigid image registration approach.

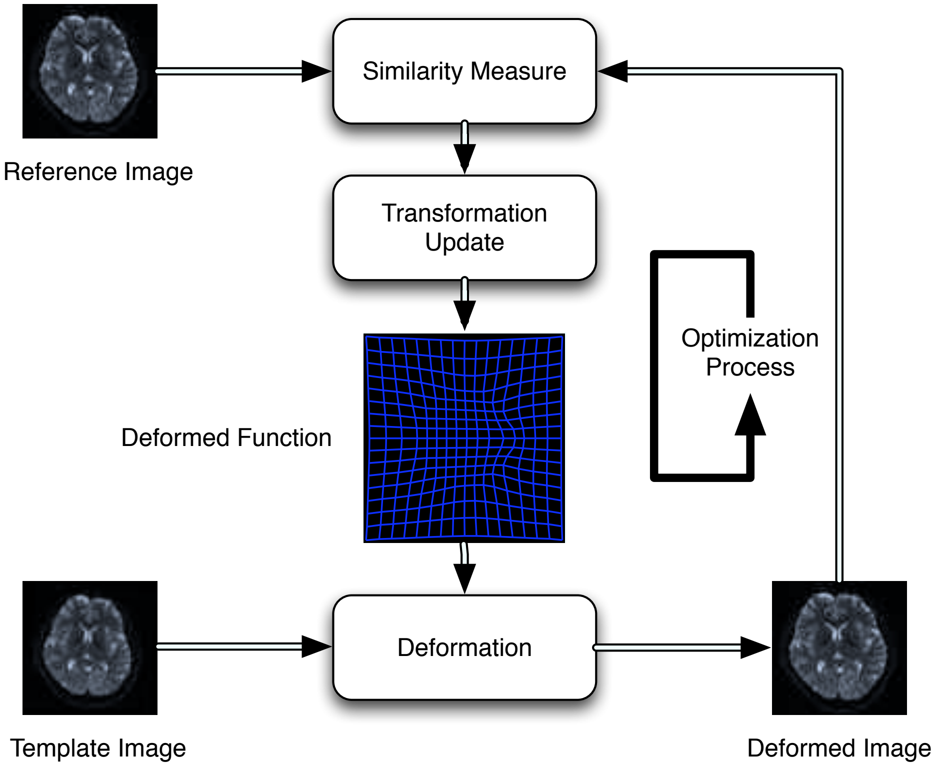


Figure 1.1. Flowchart of general non-rigid image registration approach.

There are two popular non-rigid image registration approaches, variational and parametric approaches. The idea for variational approach is to model the non-rigid image registration as a deformation process of certain material driven by external forces. Two famous methods, elastic image registration [8], and viscous fluids image registration [9] fall into this category. They both use physically-based numerical methods to model the transformation driven by the applied forces. For image registration, the driving forces are derived from the similarity measure of images. Usually, this type of registration can be formulated as a minimization problem which aims at minimizing the weighted sum of a similarity functional $\mathcal{S}$ and a regularization functional $\mathcal{R}$.

$$\mathcal{T}(\varphi) = \mathcal{S}(R, T_{\varphi}) + \alpha \mathcal{R}(\varphi), \tag{1.1}$$

where α is a positive regularizing parameter. The purpose of the regularization functional is to maintain the quality of the deformation field such that the deformation is an injective mapping. In parametric non-rigid image registration, the transformation is represented in terms of some parameters of basis functions. The most famous method of this category is the fast parametric elastic image registration [10] proposed by Kybic. In that work [10], the parameters are the B-spline coefficients [11, 12] of the displacement field at each control grid point. The complete displacement field is then obtained through B-spline interpolation. As described in [10], this method is very likely to result in grid folding in the final deformation field if the distance between adjacent control grid points (knot spacing) is less than 8. This implies that the high frequency components in the deformation field can not be accurately estimated.

Another related work was proposed by Cuzol in [13]. In that work, the deformation is considered as the result of two types of particles, namely the vortex particles and sink and source particles. The vortex particles are responsible for the vorticity (curl) of the deformation field while the sink and source particles are responsible for the divergence of the deformation field. Three types of parameters are associated with each particle: 1. the vorticity (curl) or divergence strength, 2. the influence domain, 3. position. The positions of those particles are also determined by the optimization process. Their method leads to a very complicated mathematic model and cannot be generalized to 3-D case easily. The detail of this work will be discussed on Chap. 2.

Motivated by both [10] and [13], a novel parametric non-rigid image registration algorithm based on the Helmholtzs theorem is developed in this dissertation.

Helmholtz's theorem states that a vector field is determined to within an additive constant if both its divergence and curl are specified everywhere [14]. Instead of the displacements of regular control grid points, the divergence and curl of the deformation field at each grid point are employed as the parameters. Unlike [10], where each parameter has local effect on the deformation field, the parameters used in the proposed work, the divergence and curl of the deformation field, are known to have global effects on the deformation field. Therefore it is interesting to compare their performance in terms of accuracy and robustness. In addition, unlike [13], where the positions of the vortex and source and sink particles are unknown at the beginning of the registration process. The divergence and curl, used as the only control parameters, are associated with each grid point. This leads to a very simple mathematical model and can be applied to 3D case easily.

The organization of this dissertation follows. In chapter 2, fast parametric elastic image registration methods and a parametric method for non-regid image registration based on Helmholtz's decomposition will be discussed. In chapter 3, the proposed method - Helmholtzs theorem based parametric non-rigid image registration will be described. Experimental results are presented in section 4 followed by a discussion and summary in section 5.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a parametric based non-rigid image registration named fast parametric elastic image registration [10], and a non-rigid image registration method based on Helmholtz decomposition [13] are reviewed.

2.1 Fast parametric elastic image registration

Fast parametric elastic image registration [10] is a popularly used parametric non-rigid image registration method. The basic idea is to represent the deformation field by a set of B-spline coefficients which form the set of parameters. In a multi-resolution fashion, the coefficients are associated with the nodes of grids of different scales. In 2-D, each grid point is associated with two B-spline coefficients, $c_x(\mathbf{i})$ and $c_y(\mathbf{i})$, as the parameters. The goal is to control the parameters to get a deformation field φ which aligns the reference image R and the warped template image $T_w = T^c(\varphi(\mathbf{x}))$. The notation T^c represents the continues version of the template image T whose domain is discrete by nature. In other words the intensity values of T are only defined on the regular grid points; while in the continues version T^c, the intensity values at non-grid points are well defined. In [10], the registration problem is defined as the following minimization process.

$$\mathbf{C}^* = \mathbf{arg\,min}_{\varphi \in \psi} \mathbf{ssd}(R, T, \varphi_{\mathbf{c}}) \tag{2.1}$$

where **ssd** represents the sum of squared difference metric, R and T are the two image to be registered, ψ is the space of deformation field and φ is a specific deformation field characterized by the parameter set $\mathbf{C}$. The goal is to find the parameter set $\mathbf{C}^*$ that minimizes the **ssd** similarity metric

In [10], ssd similarity metric is calculated as,

$$\begin{aligned}\mathbf{ssd}(R,T,\varphi_{\mathbf{c}}) &= \frac{1}{\|I\|}\sum_{\mathbf{i}\in I}(R(\mathbf{i})-T_w(\mathbf{i}))^2\\ &= \frac{1}{\|I\|}\sum_{\mathbf{i}\in I}(R(\mathbf{i})-T^c(\varphi_{\mathbf{c}}(\mathbf{i})))^2\end{aligned} \tag{2.2}$$

where $\mathbf{i}$ is the coordinates of each pixel/voxel, I is the set of pixels/voxels in the overlap of the images R and T, and $\|I\|$ means the number of elements in I.

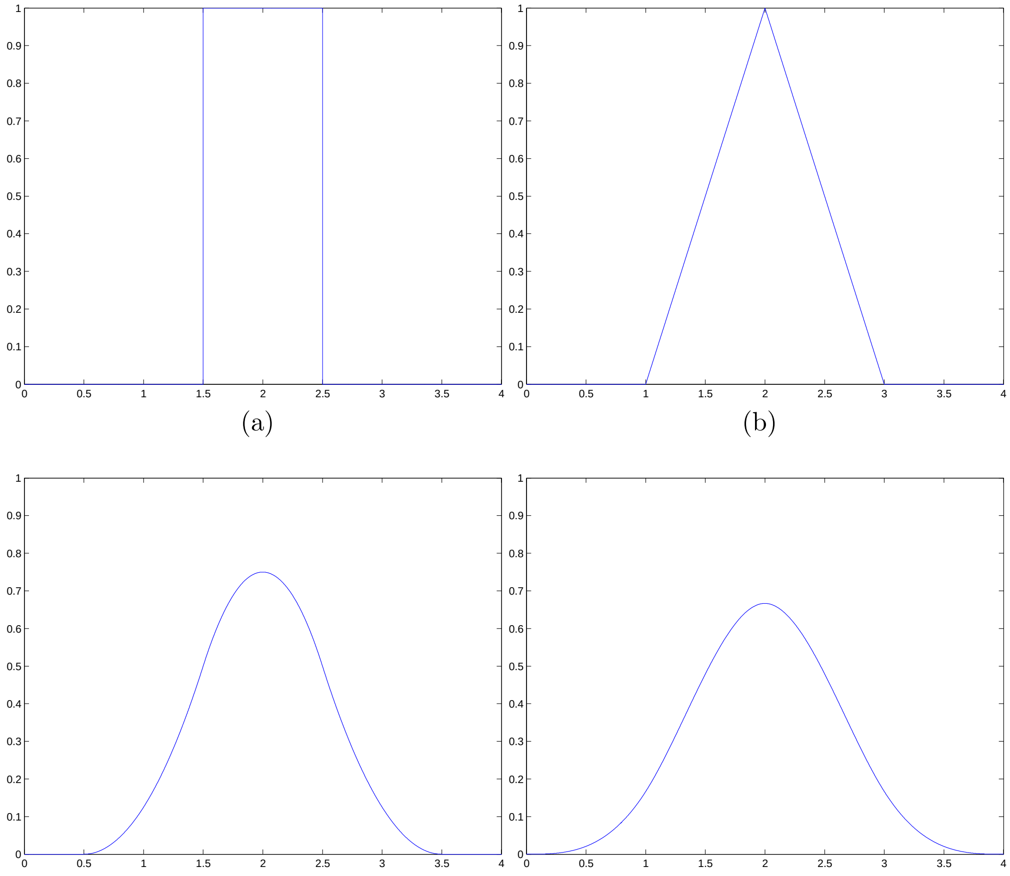
The continuous version of the template image T is obtained by using B-spline interpolation,

$$T^c(\mathbf{x}) = \sum_{\mathbf{i}\in I} b_{\mathbf{i}}\beta_n(\mathbf{x}-\mathbf{i}) \tag{2.3}$$

where $b_{\mathbf{i}}$s are the B-spline coefficients [11, 12] of the template image T, $\beta_n(\mathbf{x}) = \prod_{k=1}^{N}\beta_n(x_k)$ is a tensor product of B-spline of degree n, and N is the dimensional index which equals 2 or 3. In [10], the third order B-spline (n=3) was used. The shape of B-spline functions of different orders are shown in Fig. 2.1. Notice that, in this formulation, the cost comprises only one similarity term. In [10], the deformation field is again modeled using the third order B-spline as,

$$\varphi(\mathbf{x}) = \mathbf{x} + \sum_{\mathbf{i}\in I_c} c_{\mathbf{i}}\beta_3(\frac{\mathbf{x}}{h}-\mathbf{i}) \tag{2.4}$$

where $c_{\mathbf{i}}$s are the control parameters (B-spline coefficients) and h is the knot spacing. The regularity is enforced implicitly by using the third order B-spline interpolation that introduces the smoothness of the deformation field to a certain degree.



(a) (b)

(c) (d)

Figure 2.1. B-spline functions of degrees (a) order 0 (b) order 1 (c) order 2 (d) order 3.

In [10], four optimization methods were evaluated and the gradient descent based algorithm was suggested. The gradients can be derived by using chain rules as,

$$\frac{\partial \mathbf{ssd}}{\partial c^m(\mathbf{i})} = \frac{1}{\|I\|} \sum_{\mathbf{j} \in I_N(\mathbf{i})} \frac{\partial \mathbf{ssd}}{\partial \varphi^m(\mathbf{j})} \frac{\partial \varphi^m(\mathbf{j})}{\partial c^m(\mathbf{i})} \tag{2.5}$$

$$= \frac{1}{\|I\|} \sum_{\mathbf{j} \in I_N(\mathbf{i})} \frac{\partial \mathbf{ssd}}{\partial T_w(\mathbf{j})} \frac{\partial T^c(\varphi(\mathbf{j}))}{\partial \varphi^m(\mathbf{j})} \frac{\partial \varphi^m(\mathbf{j})}{\partial c^m(\mathbf{i})} \tag{2.6}$$

where $I_N(\mathbf{i})$ denotes some neighborhood of the grid point $\mathbf{i}$, whose size is determined by the support of the B-spline function adopted, and m is the dimensional index.

At each iteration, the set of parameters is iteratively updated based on the gradient information obatined:

$$^{i}c = {}^{i-1}c + \mu * {}^{i}\Delta c \tag{2.7}$$

where μ is the step size, $^{i}\Delta c$ is the gradient obatined in the ith iteration. In [10], the step size is increased by 10 times if the ssd value is improved, otherwise, it is decreased by a factor of 15.

The stop criterion of the optimization process is the following. If the step size μ falls below to a certain threshold ε, then the optimization process is stopped. ε is suggested to be ranged from 10^{-1} at the coarsest level to 10^{-3} at the finest level.

In the experiments of [10], multiresolution strategy is used for both the image and the deformation model. The registration process starts at the coarsest resolution of both images and refine the image and the deformation model alternatively through the finest level. It is concluded in [10] that the knot spacing h cannot be less than 8, otherwise, grid folding may result.

A useful extension,which is also adopted in this work, was developed in [10]. The extension exploits the landmark information supplied by the user to increase the robustness of the registration process. In the optimization processing, it is possible to converge to a local minimum rather than the global minumum. To avoid this situation, a second term $\mathbf{D}_{landmark}$ is added to the minimization formulation such that the cost $= \mathbf{ssd} + \mathbf{D}_{landmark}$. The term $\mathbf{D}_{landmark}$ is calculated based on a set of landmark points which are supplied by the user, and is devised as.

$$\mathbf{D}_{landmark} = \sum_{i=1}^{n} \alpha_i \|\varphi(\mathbf{x_i}) - \mathbf{z_i}\|^2 \tag{2.8}$$

where n is the number of landmark points, α_i are the weights which control the influence of each pair of landmark points. x_is are the landmark positions in the reference image and z_is are the corresponding landmark position in the template image. It is mentioned that the weights α_is should start with 1.0 and are adjusted to obtain the most satisfactory results. Too strong or weak of α_i will not lead to the correct convergence.

The flowchart of the fast parametric elastic image registration in 2-D is shown in Fig. 2.2. The performance of this method is compared with the performance of the proposed method in Chap. 5.

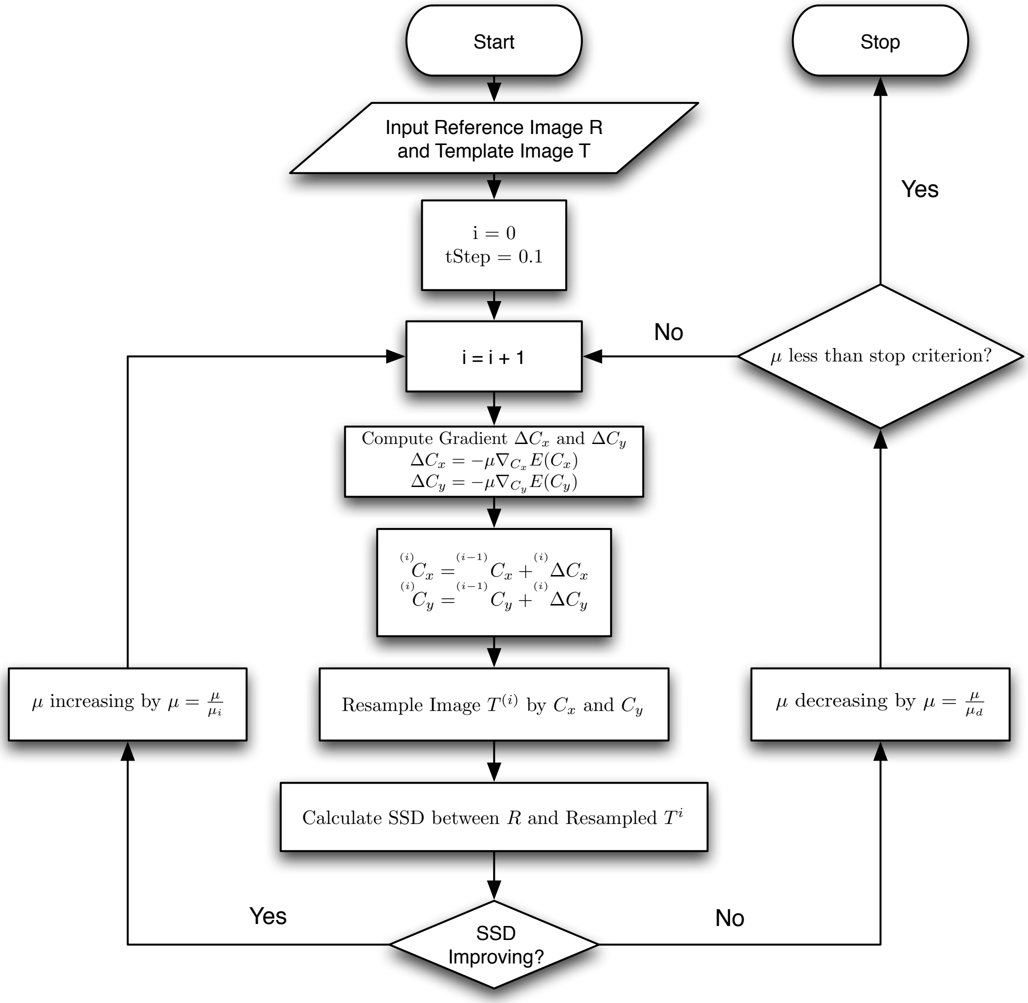


Figure 2.2. Flowchart of Fast Parametric Elastic Image registration.

2.2 A Parametric Non-Rigid Image Registration based on Helmholtz's Decomposition

The idea of using the div and curl of the deformation field as control parameters have been explored before in [13]. However, the realization of the idea is very complicated and can not be generalized to 3D easily. The details of the algorithm is provided in this chapter as a comparison to the work developed in this dissertation.

The method presented in [13] is based on Helmholtz's decomposition theorem. The theorem states that any sufficiently smooth vector field rapidly decay at infinity can be resolved into irrotational (curl-free) and solenoidal (divergence-free) component vector fields, i.e., given a vector field ϕ, based on the Helmholtz's decomposition, it can be decomposed into ϕ_i (curl free) and ϕ_s (divergence free) such that,

$$\begin{cases} \nabla \cdot \phi_i = \nabla \cdot \phi \\ \nabla \times \phi_i = 0 \end{cases} \quad \mathit{and} \quad \begin{cases} \nabla \cdot \phi_s = 0 \\ \nabla \times \phi_s = \nabla \times \phi \end{cases} \tag{2.9}$$

An additional component, laminar vector field, exists in Helmholtz's decomposition. This component only exists when the boundary condition is not null. To eliminate this component, one can perform an rigid image registration before the non-rigid image registration process. Thus, the null boundary condition can be safely applied, and the laminar component does not exist anymore.

Based on the Helmholtz's decomposition, we can sum ϕ_i and ϕ_s to obtain a div-curl system in 2D, i.e., $\phi = \phi_i + \phi_s$

$$\begin{aligned} \operatorname{div} \phi &= \operatorname{div} \phi_i = \tfrac{\partial \phi_{ix}}{\partial x} + \tfrac{\partial \phi_{iy}}{\partial y} = \nabla \cdot \phi_i \\ \operatorname{curl} \phi &= \operatorname{curl} \phi_s = -[\tfrac{\partial \phi_{sx}}{\partial y} - \tfrac{\partial \phi_{sy}}{\partial x}] = -\nabla \cdot \phi_s^{\perp} \end{aligned} \tag{2.10}$$

where $\phi = (\phi_x, \phi_y)$ and $\phi^{\perp} = (-\phi_y, \phi_x)$. Eq. (2.10) utilized the fact that $\nabla \times \phi_s = -\nabla \cdot \phi_s^{\perp}$. In order to show it is sufficiently to use $\phi_s^{\perp}$ to substitute ϕ_s, we need to maintain the property of ϕ_s in Eq. (2.9). To this end, we need to show $\nabla \times \phi_s^{\perp} = 0$.

$$\begin{aligned}\nabla \times \phi_s^{\perp} &= \frac{\partial \phi_{sx}}{\partial x} - (-\frac{\partial \phi_{sy}}{\partial y}) \\ &= \nabla \cdot \phi_s\end{aligned}$$

By the definition of ϕ_s in Eq. (2.10), $\nabla \times \phi_s^{\perp} = 0$.

There are two well known scalar fields in irrotational and solenoidal fields, Ψ and ψ, such that $\phi_i = \nabla\Psi$ and $\phi_s^{\perp} = \nabla\psi$. Ψ and ψ are known as velocity potential and stream function respectively [13]. The Eq. (2.9) can be rewritten into two Poisson equations.

$$\begin{aligned}\text{div}\,\phi_i \quad &= \nabla\nabla\Psi = \Delta\Psi \\ \text{-curl}\,\phi_s \quad &= \nabla\nabla\psi = \Delta\psi\end{aligned} \tag{2.11}$$

According to the generalized Gauss theorem [15], let $G(x)$ be the fundamental solution of above Poisson equation, where

$$G(x) = \frac{1}{2\pi} ln(\parallel x \parallel) \tag{2.12}$$

then, the solution of Eq. (2.11) may be expressed as following

$$\begin{aligned}\Psi &= \textstyle\int G(x - x^{'})\,\text{div}\,\phi_i(x^{'})dV^{'} \\ \psi &= -\textstyle\int G(x - x^{'})\,\text{curl}\,\phi_s(x^{'})dV^{'}\end{aligned} \tag{2.13}$$

Eq. (2.13) can be viewed as a convolution process.

$$\begin{aligned}\Psi &= G \otimes div\phi_i \\ \psi &= -G \otimes curl\phi_s\end{aligned} \tag{2.14}$$

By the definition of ϕ, Eq. (2.14) can be rewritten as

$$\begin{aligned}\phi_i &= \nabla\Psi = G' \otimes div\phi_i \\ \phi_s &= \nabla\psi = -G' \otimes curl\phi_s\end{aligned} \tag{2.15}$$

where

$$G' = \frac{\mathbf{x}}{2\pi \parallel \mathbf{x} \parallel^2} \tag{2.16}$$

Both the div ϕ and curl ϕ can be approximated by the source/vortex particles methods. Therefore, the deformation model can be constructed to obtain ϕ_s and ϕ_i as:

$$\begin{aligned} \phi_s(\mathbf{x}) &= \textstyle\sum_{k=0}^{n^s} \gamma_k^s \frac{(\mathbf{z}_k^s - \mathbf{x})^{\perp}}{2\pi|\mathbf{x}-\mathbf{z}_k^s|^2}(1 - \exp^{-\frac{|\mathbf{x}-\mathbf{z}_k^s|^2}{(\epsilon_k^s)^2}}) \\ \phi_i(\mathbf{x}) &= \textstyle\sum_{k=0}^{n^i} \gamma_k^i \frac{(\mathbf{x}-\mathbf{z}_k^i)}{2\pi|\mathbf{x}-\mathbf{z}_k^i|^2}(1 - \exp^{-\frac{|\mathbf{x}-\mathbf{z}_k^i|^2}{(\epsilon_k^i)^2}}) \end{aligned} \tag{2.17}$$

where n represents the number of vortex particles, γ_k^s and γ_k^i represents the strength of the the vortex particles k and source and sink particles k respectively, $\mathbf{z}_k^s$ and $\mathbf{z}_k^i$ represents the location of the vortex particles k and source and sink particles k respectively, and ϵ represents the influence domain of the smooth kernel G.

To apply above technique based on Helmholtz's decomposition to image registration problems, the positions of the source/vortex particles $(\mathbf{z}_k^s, \mathbf{z}_k^i)$, the influence domains $(\epsilon_k^s, \epsilon_k^i)$, as well as their strengths (γ_k^s, γ_k^i) are employed as the the parameters in [13]. In this manner, image registration now becomes a minimization problem. The cost function to be minimized in [13] is devised based on the linearized version of the usual brightness consistency equation:

$$\nabla I(\mathbf{x}, t)^T \cdot \phi(\mathbf{x}) + I_t(\mathbf{x}, t) = 0 \tag{2.18}$$

where $\nabla I(\mathbf{x}, t)$ represents the spatial gradient of the luminance function I, $I_t(\mathbf{x}, t)$ represents the temporal gradient.

It is mentioned in [**?**] that this optical flow constraint is not valid in cases of large displacements, thus placing another limit for this approach. Assuming the optical flow constraint is valid almost everywhere on the whole image plane, the cost function to be minimized then is devised as the following:

$$F(I, \phi) = \int_{\Omega} [\nabla I(\mathbf{x}, t)^T \cdot \phi(\mathbf{x}) + I_t(\mathbf{x}, t)]^2 dx \tag{2.19}$$

where $\phi(\mathbf{x})$ is the motion/deformation field and following minimization problem is formed,

$$\hat{\beta} = \mathbf{argmin}_{\beta} F(I, \phi(\beta)) \tag{2.20}$$

with $\beta = (\{\mathbf{z}_k^s, \gamma_k^s, \epsilon_k^s\}, \{\mathbf{z}_k^i, \gamma_k^i \epsilon_k^i\})$

The particular form of the field components Eq. (2.17) makes this a difficult minimization problem. In [13], a minimization approach based on a least square process embedded in a milti-resolution scheme and associated to a generalized conjugated gradient optimization known as Fletcher-Reeves method was developed. For interested readers, please see [13] for more details about this particular minization scheme.

CHAPTER 3

THE PROPOSED PARAMETRIC NON-RIGID IMAGE REGISTRATION USING HELMHOLTZ'S THEOREM

3.1 Helmholtz's Theorem

Helmholtz's theorem states that a vector field (vector point function) is determined to within an additive constant if both its divergence and curl are specified everywhere [14]. The divergence and curl of a vector field are defined as follows.

$$\begin{aligned} \operatorname{div}\phi &= \tfrac{\partial\phi_x}{\partial x}+\tfrac{\partial\phi_y}{\partial y} &= \nabla\cdot\phi \\ \operatorname{curl}\phi &= \tfrac{\partial\phi_y}{\partial x}-\tfrac{\partial\phi_x}{\partial y} &= \nabla\times\phi \end{aligned} \tag{3.1}$$

in two-dimensional space and

$$\begin{aligned} \nabla\cdot\phi = &\ \tfrac{\partial\phi_x}{\partial x}+\tfrac{\partial\phi_y}{\partial y}+\tfrac{\partial\phi_z}{\partial z} \\ \nabla\times\phi = &\ (\tfrac{\partial\phi_z}{\partial y}-\tfrac{\partial\phi_y}{\partial z})\hat{x} \\ &+(\tfrac{\partial\phi_x}{\partial z}-\tfrac{\partial\phi_z}{\partial x})\hat{y} \\ &+(\tfrac{\partial\phi_y}{\partial x}-\tfrac{\partial\phi_x}{\partial y})\hat{z} \end{aligned} \tag{3.2}$$

in three-dimensional space.

In an unbounded region, both the divergence and the curl of the vector field are assumed to be vanished at infinity. While in a bounded region, suitable boundary conditions are required to uniquely determine the vector field. Helmholtzs theorem can be proved as a mathematical theorem [14].

3.2 Proposed Non-rigid Image Registration Method

In this dissertation, we apply the Helmholtzs theorem to non-rigid image registration problems. The idea is to use the divergence and curl of the deformation field at regular grid points as the control parameters. Notice the difference between this approach and the approaches reviewed in Chapter 2.1 (fast parametric elastic image registration method [10]) and Chapter 2.2 (a novel parametric method for non-rigid image registration [13]). In [10], the control paprameters are the displacements at the regular grid points while in [13], the control parameters are the positions, influence domains, and strengthes of the sink and source particles and vortex particles.

3.2.1 A Basic Scheme

Let $R(\xi)$ and $T(\xi)$be two d-dimensional discrete images, $d = 2$ or 3, where $\xi \in I \subset Z^d$ is the coordinate vector of a point in the discrete domain I. We name R the reference image and T the template image, which is a geometrically distorted version of an otherwise ideally registered image $T^{ideal}(\xi)$. The goal is to find a deformation field $\varphi : \ \xi \rightarrow \xi + \phi(\xi; \mathbf{f})$, where ϕ is a displacement field, characterized by the parameter vector $\mathbf{f}(\mathbf{f} = \{\nabla \cdot \phi, \nabla \times \phi\})$, such that a valid similarity measure S is optimized, i.e.,

$$\mathbf{f} = \arg\operatorname{opt} S(R(\xi), T(\varphi(\xi, \mathbf{f}))) \tag{3.3}$$

In this formulation, non-rigid image registration becomes a parametric optimization problem comprising only the similarity term. The parameters (div and curl values) are associated with each pixel/voxel. The required regularity is implied in the underlying div-curl representation of the displacement field. That is, the displacement field is always one order smoother than the divergence and curl fields. This is clearly seen from Eq. (3.1) and (3.2). To reduce the number of parameters, a coarse grid $I^h \subset I$

can be utilized whose grid points are where the parameters are associated with, and h is the knot spacing. The complete displacement field is then computed from the displacement field on the coarse grid through interpolation.

In order to solve our minimization problem, two major components have to be addressed. The first one is an efficient div-curl solver, and the second one is a gradient-based optimizer. In the next section, a div-curl solver based on a set of Poisson's equations is presented.

3.2.2 Div-Curl Solver

According to Helmholz's theorem, a displacement field required in non-rigid image registration can be represented in two different ways: a direct representation of the the displacement defined at each pixel/voxel, or, an indirect representation through the divergence and curl of the displacement filed with suitable boundary conditions. Given a displacement field, it is easy to get the div-curl representation through equation (3.1) or (3.2). However, the reverse process is not trivial and requires an efficient numerical div-curl solver. In this work, we presented a div-curl solver by transforming a div-curl system into d ($d = 2$ or 3) Poisson's equations, which can then be solved using various finite difference methods [16]. The derivation in both 2D and 3D cases are presented next.

In 2D case, let

$$\mathbf{div}\phi = \frac{\partial \phi_x}{\partial x} + \frac{\partial \phi_y}{\partial y} = f^1 \tag{3.4}$$

$$\mathbf{curl}\phi = \frac{\partial \phi_y}{\partial x} - \frac{\partial \phi_x}{\partial y} = f^2 \tag{3.5}$$

We assume that f^i, $i = 1{,}2$ are at least C_1 continuous. Taking the derivative of both sides of each equation with respect to x, and y and combine the relevant terms, we arrive at the following two Poisson's equations,

$$\begin{aligned} \Delta\ \phi_x &= f^1_x - f^2_y \equiv F^1 \\ \Delta\ \phi_y &= f^1_y + f^2_x \equiv F^2 \end{aligned} \tag{3.6}$$

where

$$f^i_k = \frac{\partial f^i}{\partial k} \tag{3.7}$$

In 3D case, let

$$\mathbf{div}\phi = \frac{\partial \phi_x}{\partial x} + \frac{\partial \phi_y}{\partial y} + \frac{\partial \phi_z}{\partial z} = f^1 \tag{3.8}$$

$$\mathbf{curl}_x\phi = \frac{\partial \phi_z}{\partial y} - \frac{\partial \phi_y}{\partial z} = f^2 \tag{3.9}$$

$$\mathbf{curl}_y\phi = \frac{\partial \phi_x}{\partial z} - \frac{\partial \phi_z}{\partial x} = f^3 \tag{3.10}$$

$$\mathbf{curl}_z\phi = \frac{\partial \phi_y}{\partial x} - \frac{\partial \phi_x}{\partial y} = f^4 \tag{3.11}$$

We assume that f^i, $i = 1{,}2{,}3{,}4$ are at least C_1 continuous. Taking the derivative of both sides of each equation with respect to x, y, and z and combine the relevant terms, we arrive at the following three Poisson's equations,

$$\begin{aligned} \Delta\ \phi^x &= f^1_x +\ f^3_z - f^4_y \equiv F^1 \\ \Delta\ \phi^y &= f^1_y +\ f^4_x - f^2_z \equiv F^2 \\ \Delta\ \phi^z &= f^1_z +\ f^2_y - f^3_x \equiv F^3 \end{aligned} \tag{3.12}$$

where

$$f^i_k = \frac{\partial f^i}{\partial k} \tag{3.13}$$

In most cases including ours, these equations cannot be solved analytically, and therefore we must use try the numerical methods. The Poisson's equations can be solved in several different ways. In the following sections, four numerical Poisson solvers are discussed - direct method, iterative based, inverse filter based, and fast Fourier transform based. Since all of these may contain numerical errors, we would like to know how the errors affect our registration method.

3.2.2.1 Direct Method

Let's discretize the 2D Poisson equation using Dirichilet boundary condition with boundary values being zero,

$$\begin{aligned} &\Delta\phi = F \\ \Rightarrow &\frac{d^2\phi}{dx^2} + \frac{d^2\phi}{dy^2} = F \end{aligned} \tag{3.14}$$

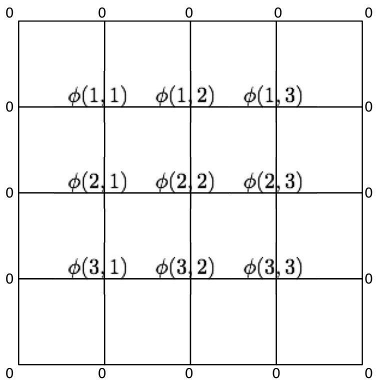
The discretized form of Eq. (3.14) is

$$\frac{\phi(i-1,j) - 2\phi(i,j) + \phi(i+1,j)}{h^2} + \frac{\phi(i,j-1) - 2\phi(i,j) + \phi(i,j+1)}{h^2} = F(i,j) \tag{3.15}$$

where (i,j) represents the coordinate of the grid point, and h represents the knot spacing.

To facilitate the representation of the direct method, in what follows, we assume the discrete domain of our problem is as shown below

Then, Eq. (3.15) can be rewritten in matrix form as following.



$$\begin{array}{ccccc}
 & K & \Phi & & F \\
\frac{1}{h^2} &
\begin{bmatrix}
-4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4
\end{bmatrix}
&
\begin{bmatrix}
\phi(1,1) \\ \phi(2,1) \\ \phi(3,1) \\ \phi(1,2) \\ \phi(2,2) \\ \phi(3,2) \\ \phi(1,3) \\ \phi(2,3) \\ \phi(3,3)
\end{bmatrix}
& = &
\begin{bmatrix}
F(1,1) \\ F(2,1) \\ F(3,1) \\ F(1,2) \\ F(2,2) \\ F(3,2) \\ F(1,3) \\ F(2,3) \\ F(3,3)
\end{bmatrix}
\end{array} \tag{3.16}$$

where K represents the Laplacian operator in matrix form. Since K is a squared matrix and has full rank, the inverse of K can be computed directly. This method is able to solve the Poisson equation quite accurately, but it also demands a significant memory space to store the matrix K and calculate the inverse matrix of K. The size of matrix K grows rapidly as the size of image increases. If a given image has $n \times n$ pixels, then the size of matrix K is $(n \times n)^2$. In Matlab, each matrix element uses 8 bytes. If n = 257, it takes about 4 giga bytes to store one K matrix. Also, the complexity of direct method is around $O(n^2)$, which is formidable. Next we review an iterative Poisson solver that does not demand large memory space.

3.2.2.2 Successive Over Relaxation Method

This simple and straight-forward method [17] is often the first choice for practically solving the Poisson equation. It provides an approximative solution based on the Gauss-Seidel method, which is modified by adding a parameter α to accelerate the convergence. The Gauss-Seidel method is an iterative method to solve a linear system of equations. To solve Poisson equation $K\Phi = F$, where K is a m-by-m Laplacian operator matrix, and Φ is a m-by-1 matrix, $\phi \in \Phi$. The Gauss-Seidel iterative process can be expressed as follows with the initial values of Φ are zero:

$$\Phi_i^{(r+1)} = \frac{1}{K_{(i,i)}} \left[F_i - \sum_{j=1}^{j<i} K_{(i,j)} \Phi_j^{(r+1)} - \sum_{j>i}^{m} K_{(i,j)} \Phi_j^{(r)} \right] \tag{3.17}$$

where (i, j) represents coordinate index which is from $1, 2, ..., m$, r represents the iteration index. In this iterative process, each iterative step consists of the update performed on each of the 2D grid nodes inside the computational domain. However, the convergence rate is slow. SOR method puts a weight between current solution and previous solution to speed up the convergence. That is,

$$\Phi_i^{(r+1)} = (1-\alpha)\,\Phi_i^{(r)} + \alpha \left\{ \frac{1}{K_{(i,i)}} \left[F_i - \sum_{j=1}^{j<i} K_{(i,j)} \Phi_j^{(r+1)} - \sum_{j>i}^{m} K_{(i,j)} \Phi_j^{(r)} \right] \right\} \tag{3.18}$$

To increase the rate of convergence, a good initial guess is desirable. The successive over relaxation method can be used to solve the Poisson's equations formed by a div-curl system, and it does not need extra memory storage. However, compare to other efficient methods, the complexity of SOR $(O(n^2))$ is still too high.

3.2.2.3 Inverse Filter Based Div-Curl Solver

The inverse filter based div-curl solver is devised for two reasons. First of all, it is scalable, meaning that there is a trade-off between the accuracy and speed. Secondly, it provides a means to derive the gradient information required in the optimization process which is otherwise difficult to attain.

Before we present the inverse filter based Poisson solver, two convolution operations that will be used in this section need to be defined. Suppose we would like to perform the following operation

$$m \star A = B \tag{3.19}$$

where $\star$ is one of the following convolution operations: $*$, and $*_s$, m is a 3-by-3 filter with indices ranging from -1 to 1 and A is a matrix of size m-by-n then
if $\star = *$, it performs a traditional convolution

$$B(i,j) = \sum_{a=-1}^{1} \sum_{b=-1}^{1} m(a,b) \cdot \hat{A}(i+a, j+b) \tag{3.20}$$

where

$$0 \le i \le m+1$$
$$0 \le j \le n+1$$

$$\hat{A} = \begin{cases} A(i,j) & \text{if} \quad \begin{matrix} 1 \le i \le m \\ 1 \le j \le n \end{matrix} \\ 0 & \text{otherwise} \end{cases}$$

if $\star = *_s$, it performs a convolution operation which will return only the central part of the results of the traditional convolution operation which has the same size as the operand which has the larger size.

$$B(i,j) = \sum_{a=-1}^{1} \sum_{b=-1}^{1} m(a,b) \cdot \hat{A}(i+a, j+b) \tag{3.21}$$

where

$$1 \leq i \leq m$$

$$1 \leq j \leq n$$

$$\hat{A} = \begin{cases} A(i,j) & \text{if} \quad \begin{matrix} 1 \leq i \leq m \\ 1 \leq j \leq n \end{matrix} \\ 0 & \text{otherwise} \end{cases}$$

Let I be the grid defining the discrete image domain. Denote $\phi(I)$ as the solution ϕ on I. Assuming null boundary condition, the Poisson's equations in (3.12) can be rewritten as convolution products after finite central difference approximation in 2D as,

$$m *_s \phi_x(I) = F^1$$

$$m *_s \phi_y(I) = F^2 \tag{3.22}$$

$$\tag{3.23}$$

where m is the 2D discrete Laplacian operator given below.

$$m = \begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array} \tag{3.24}$$

In 3D case,

$$m *_s \phi_x(I) = F^1$$

$$m *_s \phi_y(I) = F^2 \tag{3.25}$$

$$m *_s \phi_z(I) = F^3$$

where m is the 3D discrete Laplacian operator given below.

$$m\left(:,:,1\right)=\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array},\ m\left(:,:,2\right)=\begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & -6 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array},\ m\left(:,:,3\right)=\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} \tag{3.26}$$

Assume an inverse filter m^{-1} exists that satisfies

$$m *_s m^{-1} = p \tag{3.27}$$

where p represents an impulse function. Lets convolve m^{-1} with both sides of Eq. (3.25), we get

$$m^{-1} *_s m *_s \phi(I) = m^{-1} *_s F(I) \tag{3.28}$$

Finally, the solution ϕ can be obtained using inverse filtering as

$$\phi\left(I\right) = m^{-1} *_s F\left(I\right) \tag{3.29}$$

where m^{-1} is the inverse filter of the discrete Laplacian operator m under the operation $*_s$.

To find the inverse filter m^{-1}, we first choose the method in [18] for viscous fluid registration [19]. In [18], the original viscous fluid registration can be represented by combination of convolution operations based on a 3x3 filter. The goal is to use least mean square (LMS) error as a criterion to find an approximation of the inverse filter of the discrete Laplacian operator m, such that $m *_s m^{-1}$ is as close to an impulse function as possible. However, this method is memory demanding involving construction of a large matrix and finding the pseudo inverse of it. The memory requirements grows up exponentially as the image size increase. In our experiments, this method cannot handle the image size over 73x73 pixels (Based on a 2GB RAM computer). Thus, we devised a new method to find the inverse filter m^{-1} by using SOR method.

In Eq. (3.27), m and p are known. The linear equation (3.27) can be solved by utilizing successive over relaxation method to approximate the inverse of a given filter. To this end, Eq. (3.18) needs to be slightly modified as following.

$$\begin{aligned}(m^{-1}_{(i,j)})^{k+1} = (1-\alpha)\,(m^{-1}_{(i,j)})^{k} + \alpha \Bigg\{ \frac{1}{m_{center}} \Bigg[p_{(i,j)} - \sum_{a}\sum_{b} m_{(a,b)} \cdot (m^{-1}_{(i+a,j+b)})^{k+1} \\ - \sum_{a'}\sum_{b'} m_{(a',b')} \cdot (m^{-1}_{(i+a',j+b')})^{k} \Bigg]\Bigg\} \end{aligned} \tag{3.30}$$

where

$$(a,b) = \left\{\begin{array}{r} (-1,-1) \\ (0,-1) \\ (1,-1) \\ (-1,0) \end{array}\right\} \qquad (a^{'},b^{'}) = \left\{\begin{array}{r} (1,0) \\ (-1,1) \\ (0,1) \\ (1,1) \end{array}\right\}$$

For example, given the 3-by-3 Poisson operator m and a 3-by-3 impulse function p we want to solve for m^{-1}. Suppose zero boundary condition padded outside of m and the initial values of $m^{-1}_{(i,j)}, 0 \le i, j \le n-1$, are zero

$$\begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array} *_s \begin{array}{|c|c|c|} \hline m^{-1}_{(1,1)} & m^{-1}_{(1,2)} & m^{-1}_{(1,3)} \\ \hline m^{-1}_{(2,1)} & m^{-1}_{(2,2)} & m^{-1}_{(2,3)} \\ \hline m^{-1}_{(3,1)} & m^{-1}_{(3,2)} & m^{-1}_{(3,3)} \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

The equation of the first grid point $m^{-1}_{(1,1)}$ can be written as

$$0 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 1 \cdot 0 + (-4) \cdot m^{-1}_{(1,1)} + 1 \cdot m^{-1}_{(2,1)} + 0 \cdot 0 + 1 \cdot m^{-1}_{(1,2)} + 0 \cdot m^{-1}_{(2,2)}$$

$$= p(1,1) = 0$$

$$\Rightarrow -4m^{-1}_{(1,1)} = 0$$

$$\Rightarrow m^{-1}(1,1) = 0$$

Plug this result into Eq. (3.30) with $\alpha = 0.6$, the first element of m^{-1} will be updated as following

$$ {}^{1}m^{-1}_{(1,1)} = (1-0.6)\,{}^{0}m^{-1}_{(1,1)} + 0.6\left\{\frac{1}{m_{(2,2)}}\left[p_{(1,1)} - (m^{-1}_{(1,1)} *_c m)\right]\right\} = 0 $$

In the first iteration, the first non-zero value of m^{-1} is updated when the convolution process reach the center of p.

$$ \begin{aligned} &0 \cdot m^{-1}_{(1,1)} + 1 \cdot m^{-1}_{(2,1)} + 0 \cdot m^{-1}_{(3,1)} + 1 \cdot m^{-1}_{(1,2)} + (-4) \cdot m^{-1}_{(2,2)} + 1 \cdot m^{-1}_{(3,2)} + 0 \cdot m^{-1}_{(1,3)} \\ &+ 1 \cdot m^{-1}_{(2,3)} + 0 \cdot m^{-1}_{(3,3)} = p(2,2) = 1 \\ &\Rightarrow -4m^{-1}_{(2,2)} = 1 \\ &\Rightarrow m^{-1}_{(2,2)} = -0.25 \end{aligned} $$

Again, plug this result into Eq. (3.30), the value of $m^{-1}_{(2,2)}$ will be updated as following

$$ \begin{aligned} {}^{1}m^{-1}_{(2,2)} &= (1-0.6)\,{}^{0}m^{-1}_{(2,2)} + 0.6\left\{\frac{1}{m_{(2,2)}}\left[p_{(2,2)} - (m^{-1}_{(2,2)} \otimes_c m)\right]\right\} \\ &= 0.4 \cdot 0 + 0.6(-0.25) \\ &= -0.15 \end{aligned} $$

After the first non-zero value is updated, the rest of value of m^{-1} will keep updating until reach its convergence.

The size of inverse filter can be various, however, theoretically only when the size of the inverse filter m^{-1} is double of the size of ϕ, the solution obtained is exactly the same as the solution obtained by the direct method. It is because the boundary of $m * m^{-1}$ are numerical errors. To avoid these errors attaching on the image data, the effective domain of m^{-1} must cover the whole image which means the boundary of m^{-1} will not be able to affect the image. In order to get the exact solution of the Poisson equations by inverse filter, the correct boundary condition must be provided.

However, since our Poisson equations are composed by the combination of the gradient of the div-curl system (Eq. (3.6) and (3.12)), the boundary values are all affected by numerical errors. Thus the inverse filter can only be considered as an approximated solution in our cases. The advantages of proposed inverse filtering based method over the direct method are:

1 . The inverse filter m^{-1} under the operation $*_s$ can be calculated offline, since it has nothing to do with F and depends only on the size of $F(I)$.

2 . To speed up the calculation, the size of m^{-1} can be reduced. However, the solution is no longer as same as the solution from the direct method and is considered only an approximation.

The complexity of inverse filter method is $O(n^2)$.

3.2.2.4 Fast Fourier Transform Method

Another efficient Poisson solver is FFT based [20]. Let's rewrite Eq. (3.15) in another form as shown below,

$$T \cdot \phi + \phi \cdot T = F \tag{3.31}$$

where

$$T = \begin{bmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -2 \end{bmatrix} \tag{3.32}$$

T is the familiar symmetric tridiagonal matrix, and can be factorized to $T = Q \cdot \Lambda \cdot Q^{-1}$. Q is the eigenvector of T and the diagonal entries of Λ are eigenvalues of T. The explicit form of Q and Λ are:

$$Q(i,j) = sin(\frac{i \cdot j \cdot \pi}{n+1}) \cdot \sqrt{\frac{2}{n+1}} \tag{3.33}$$

$$\Lambda(i) = -2 \cdot (1 - cos(\frac{i \cdot \pi}{n+1})) \tag{3.34}$$

After substituting Eq. (3.33) and Eq. (3.34) into Eq. (3.31), we get

$$(Q \cdot \Lambda \cdot Q^{-1}) \cdot \phi + \phi \cdot (Q \cdot \Lambda \cdot Q^{-1}) = F \tag{3.35}$$

Multiply Q^{-1} and Q on both side of Eq. (3.35),

$$Q^{-1} \cdot [(Q \cdot \Lambda \cdot Q^{-1}) \cdot \phi + \phi \cdot (Q \cdot \Lambda \cdot Q^{-1})] \cdot Q = Q^{-1} \cdot F \cdot Q \tag{3.36}$$

By distributive of matrix multiplication,

$$Q^{-1} \cdot (Q \cdot \Lambda \cdot Q^{-1}) \cdot \phi \cdot Q + Q^{-1} \cdot \phi \cdot (Q \cdot \Lambda \cdot Q^{-1}) \cdot Q = Q^{-1} \cdot F \cdot Q \tag{3.37}$$

By associative of matrix multiplication,

$$(Q^{-1} \cdot Q) \cdot \Lambda \cdot (Q^{-1} \cdot \phi \cdot Q) + (Q^{-1} \cdot \phi \cdot Q) \cdot \Lambda \cdot (Q^{-1} \cdot Q) = Q^{-1} \cdot F \cdot Q \tag{3.38}$$

Let $\hat{\phi} = Q^{-1} \cdot \phi \cdot Q$ and $\hat{F} = Q^{-1} \cdot F \cdot Q$. Since the Λ is a diagonal matrix. The matrix $\hat{\phi}$ can be evaluated element by element,

$$\hat{\phi}(i,j) = \hat{F}/(\Lambda(i,i) + \Lambda(j,j)) \tag{3.39}$$

Finally, ϕ can be obtained by

$$\phi = Q \cdot \hat{\phi} \cdot Q^{-1} \tag{3.40}$$

To speed up the matrix multiplication operation, we can compute those Q matrix multiplication by using fast Fourier transform. The connection of these matrix multiplications and FFT is the following.

Let A an m-by-m matrix, be defined as The Discrete Fourier Transform of an m-by-1 vector b is the matrix-vector product $A \cdot b$, where A is a m-by-m matrix defined as following. Let

$$\omega(j,k) = cos(\frac{2 \cdot \pi \cdot j \cdot k}{m}) + \mathbf{i} \cdot sin(\frac{2 \cdot \pi \cdot j \cdot k}{m}) \tag{3.41}$$

For $0 \leq j, k \leq m-1$, then $A(j,k) = \omega(j,k)$.

Assume the size of Q is n-by-n and the size of A is m-by-m where $m = 2(n+1)$. Thus,

$$A(j,k) = cos(\frac{\pi \cdot j \cdot k}{n+1}) + \mathbf{i} \cdot sin(\frac{\pi \cdot j \cdot k}{n+1}) \tag{3.42}$$

Compare to Eq. (3.33), we realize Q is the imaginary part of A multiplied by $\sqrt{\frac{2}{n+1}}$. Therefore, the result of multiplication of $Q \cdot \hat{\phi}$ is equivalent to the imaginary part of $A \cdot \hat{\phi}_F$ in frequency domain where $\hat{\phi}_F = FFT([0, \hat{\phi}_F, \text{zeros}(n,1)])$. The detail can be found in [20]. The complexity of FFT method is $O(NlogN)$. Since this method always provides the exact solution of the Poisson equation under zero boundary condition, we have chosen this FFT-based Poisson solver to solve the div-curl system to obtain the displacement field. All other methods that introduced in this chapter have been implemented and embedded into our system. In chapter 5, the comparison of these div-curl solvers are provided.

3.2.3 Gradient Descent Optimization

Employing SSD as the similarity measure and discretizing the image domain, the gradient information required by a gradient descent optimizer can be derived by applying the chain rule repeatedly as (illustrated in 3D case)

$$\begin{aligned}\frac{\partial\,\mathrm{ssd}}{\partial f^J(I_i)} = &\sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_x(I_l)} \sum_{k\in N(j)} \frac{\partial \phi_x(I_l)}{\partial \phi_{x_h}(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_{x_h}(I_k)}{\partial F^1(I_j)} \frac{\partial F^1(I_j)}{\partial f^J(I_i)} \\ &+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_y(I_l)} \sum_{k\in N(j)} \frac{\partial \phi_y(I_l)}{\partial \phi_{y_h}(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_{y_h}(I_k)}{\partial F^2(I_j)} \frac{\partial F^2(I_j)}{\partial f^J(I_i)} \\ &+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_z(I_l)} \sum_{k\in N(j)} \frac{\partial \phi_z(I_l)}{\partial \phi_{z_h}(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_{z_h}(I_k)}{\partial F^3(I_j)} \frac{\partial F^3(I_j)}{\partial f^J(I_i)}\end{aligned} \tag{3.43}$$

where J = 1,...,4 representing the component of the parametric vector **f**, and $N(i)$ and $N(j)$ denote some neighborhood of grid point I_i and I_j respectively. The effect of varying F^l value at grid I_j on ϕ_{K_h}, l = 1,2,3 and K = x,y,z, determines the term $\frac{\partial \phi_{K_h}(I_k)}{\partial F^l(I_j)}$ and $N(j)$ is its influence domain. This can be established through the inverse filter m^{-1} in (3.29). Similarly, the effect of varying f^J value at grid I_i on F^l , $J = 1, 2, 3, 4$ and $l = 1, 2, 3$, determines the term $\frac{\partial F^l(I_j)}{\partial f^J(I_i)}$and its influence domain $N(i)$. This information can be obtained from (3.12). Using finite central difference method to approximate the involved derivatives, they can be represented as the following 3x3x3 filters.

$$\frac{\partial F^1}{\partial f^1}(:,:,1) = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&0&0\\\hline 0&0&0\\\hline\end{array},\quad \frac{\partial F^1}{\partial f^1}(:,:,2) = \begin{array}{|c|c|c|}\hline 0&-1&0\\\hline 0&0&0\\\hline 0&1&0\\\hline\end{array},\quad \frac{\partial F^1}{\partial f^1}(:,:,3) = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&0&0\\\hline 0&0&0\\\hline\end{array}$$

$$\frac{\partial F^2}{\partial f^1}(:,:,1) = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&0&0\\\hline 0&0&0\\\hline\end{array},\quad \frac{\partial F^2}{\partial f^1}(:,:,2) = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -1&0&1\\\hline 0&0&0\\\hline\end{array},\quad \frac{\partial F^2}{\partial f^1}(:,:,3) = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&0&0\\\hline 0&0&0\\\hline\end{array}$$

$$\frac{\partial F^3}{\partial f^1}(:,:,1) = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & -1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}, \; \frac{\partial F^3}{\partial f^1}(:,:,2) = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}, \; \frac{\partial F^3}{\partial f^1}(:,:,3) = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

$$\begin{gathered} \tfrac{\partial F^2}{\partial f^2} = -\tfrac{\partial F^3}{\partial f^1}, \tfrac{\partial F^3}{\partial f^2} = \tfrac{\partial F^2}{\partial f^1}, \tfrac{\partial F^1}{\partial f^3} = \tfrac{\partial F^3}{\partial f^1} \\ \tfrac{\partial F^3}{\partial f^3} = -\tfrac{\partial F^1}{\partial f^1}, \tfrac{\partial F^1}{\partial f^4} = -\tfrac{\partial F^2}{\partial f^1}, \tfrac{\partial F^2}{\partial f^4} = \tfrac{\partial F^1}{\partial f^1}, \tfrac{\partial F^1}{\partial f^2} = \tfrac{\partial F^2}{\partial f^3} = \tfrac{\partial F^3}{\partial f^4} = 0 \end{gathered} \tag{3.44}$$

If $h = 1$, then $\frac{\partial \phi_K}{\partial \phi_{K_h}} = 1$. Therefore, the gradient information can be computed through a series of convolutions as

$$\begin{aligned} \nabla_{f^J} ssd = \nabla_{\varphi_x} ssd *_s m^{-1} *_s D_{\frac{\partial F^1}{\partial f^J}} \\ +\nabla_{\varphi_y} ssd *_s m^{-1} *_s D_{\frac{\partial F^2}{\partial f^J}} \\ +\nabla_{\varphi_z} ssd *_s m^{-1} *_s D_{\frac{\partial F^3}{\partial f^J}} \end{aligned} \tag{3.45}$$

If the parameters are associated with a coarse grid, which means $h \neq 1$ and interpolation is used to obtain the complete deformation field, one more convolution kernel H is required in Eq. (3.45).

$$\begin{aligned} \nabla_{f^J} ssd = \nabla_{\varphi_x} ssd *_s H *_s m^{-1} *_s D_{\frac{\partial F^1}{\partial f^J}} \\ +\nabla_{\varphi_y} ssd *_s H *_s m^{-1} *_s D_{\frac{\partial F^2}{\partial f^J}} \\ +\nabla_{\varphi_z} ssd *_s H *_s m^{-1} *_s D_{\frac{\partial F^3}{\partial f^J}} \end{aligned} \tag{3.46}$$

Clearly, H depends on the interpolation scheme used to interpolate the displacement field from a coarse grid to the finest grid. A few examples of the linear interpolation kernel H_{1D} in 1D is shown below,

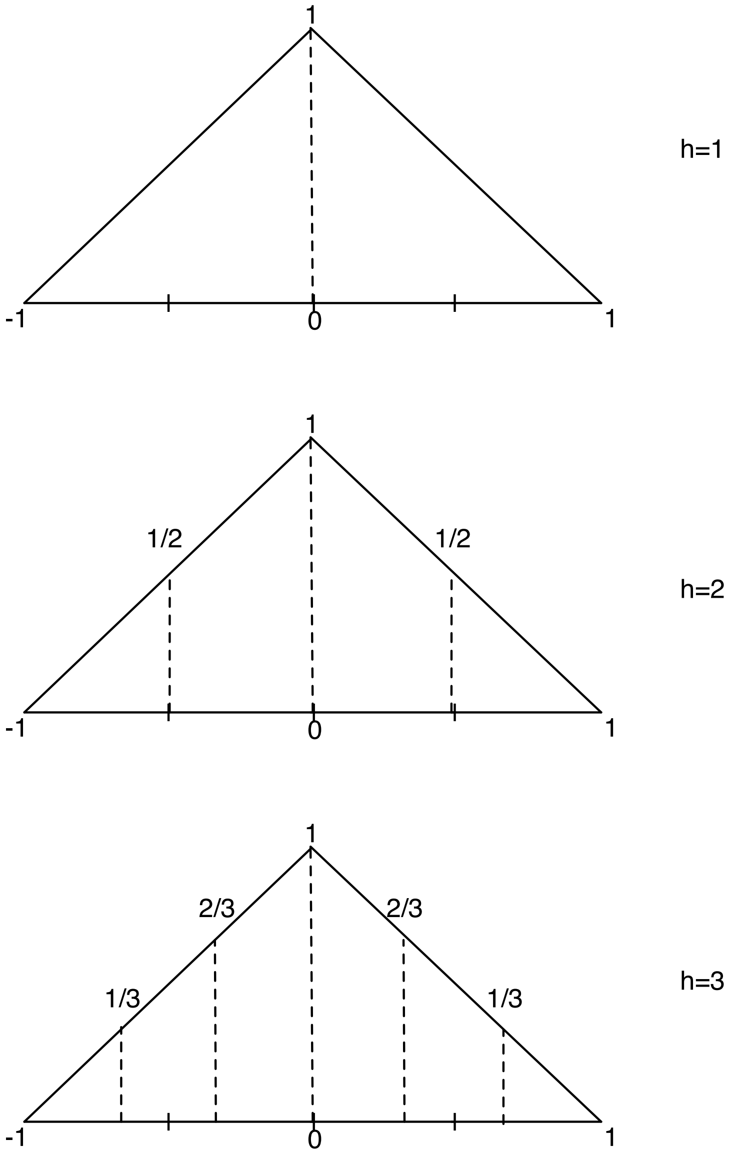


Figure 3.1. Linear Interpolation Kernel in 1D for knot spacing h = 1,2,3.

Assume $h = 2$, the value of H_{1D} is $[0, 0.5, 1, 0.5, 0]$. To obtain higher order kernel H, the tensor product is used. Suppose we would like to get H_{2D} for h=2, i.e., $H_{2D} = H'_{1D} \otimes H_{1D}$, we will get

$$H_{2D} = \begin{array}{|c|c|c|c|c|}\hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0.25 & 0.5 & 0.25 & 0 \\ \hline 0 & 0.5 & 1 & 0.5 & 0 \\ \hline 0 & 0.25 & 0.5 & 0.25 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline \end{array} \tag{3.47}$$

The 3D interpolation kernel can be derived in a similar fashion.

Once the gradient information is available, a gradient descent optimization can be devised. This gradient descent optimization method is used to optimize Eq. (3.3). The time steps are increased if the SSD improved, and decreased otherwise. The flowchart of the basic scheme of the proposed method in 2-D is shown in Fig. 3.2.

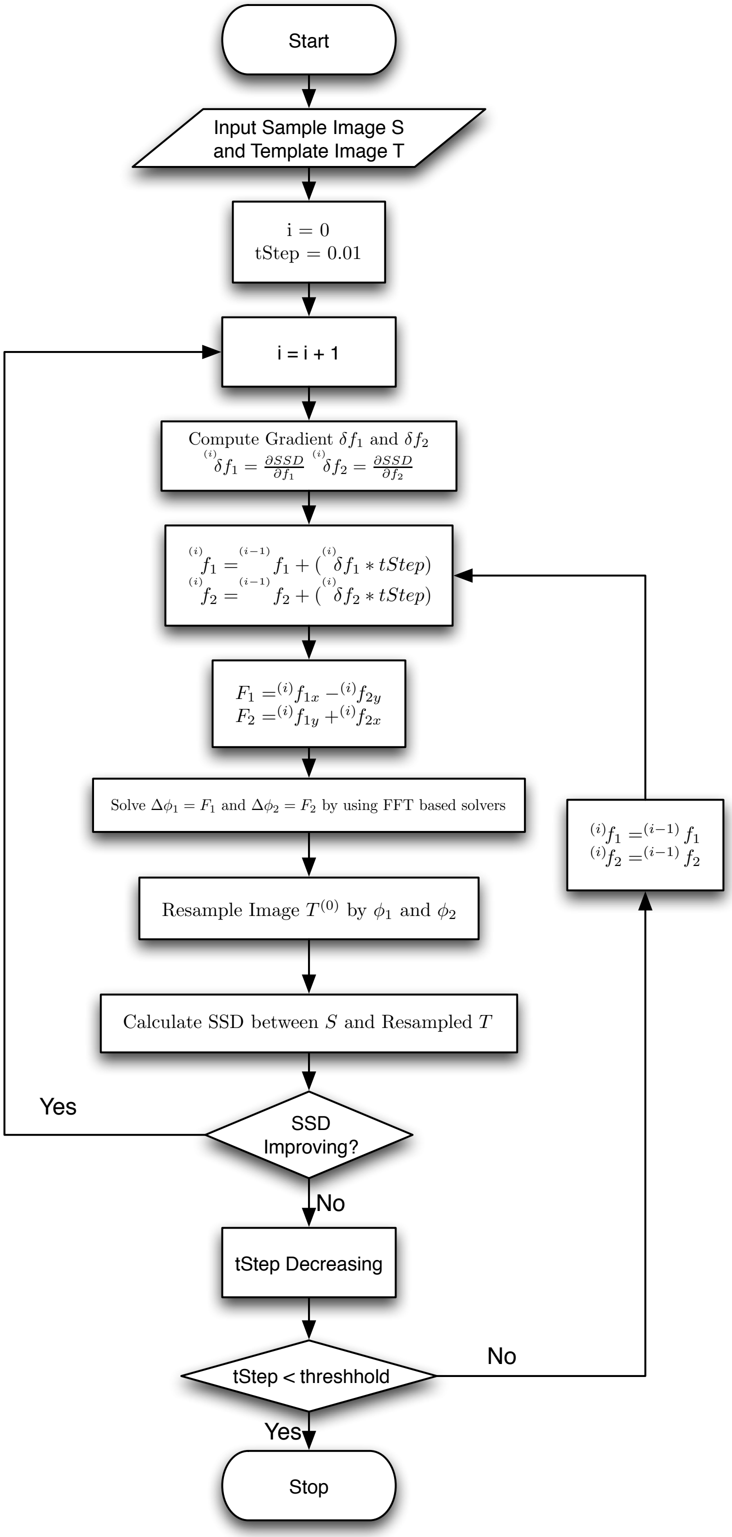


Figure 3.2. Basic scheme of the proposed method using Helmholtz's theorem.

3.3 A Simplified Scheme

In the basic scheme, we try to minimize the similarity measure with respect to the divergence and curl values. This results in two parameters with each grid point in 2D case and four parameters with each grid point in 3D case. Observing from Eq. (3.6) and Eq. (3.12), it is possible to set the F^i (in 2D, i=1,2 or in 3D, i=1,2,3) values as the parameters associated with each grid point directly. In this manner, instead of adjusting four parameters in 3D case, only three parameters are involved for each 3D grid point. In 2D case, we still adjust two parameters though, the derivation of the gradient information is simplied. This observation may result in a more efficient scheme mathematically. The idea can be expressed as to find a deformation field $\varphi :\ \xi \rightarrow \xi + \phi(\xi; \mathbf{F})$, characterized by the parameter vector $\mathbf{F}$ where $\mathbf{F}$ is defined by,

$$\begin{aligned} \Delta\, \phi_x &= F^1 \\ \Delta\, \phi_y &= F^2 \\ \Delta\, \phi_z &= F^3 \end{aligned} \tag{3.48}$$

such that a valid similarity measure S is optimized, i.e.,

$$\mathbf{F} = \arg\operatorname{opt}\, S(R(\xi), T(\varphi(\xi, \mathbf{F}))) \tag{3.49}$$

Now, the formula used to find the gradient information can be simplified as follows,

$$\begin{aligned} \frac{\partial\,\mathrm{ssd}}{\partial F^1(I_i)} &= \sum_{k\in N(j)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_x(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_x(I_k)}{\partial \phi_{x_h}(I_j)} \frac{\partial \phi_{x_h}(I_j)}{\partial F^1(I_i)} \\ \frac{\partial\,\mathrm{ssd}}{\partial F^2(I_i)} &= \sum_{k\in N(j)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_y(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_y(I_k)}{\partial \phi_{y_h}(I_j)} \frac{\partial \phi_{y_h}(I_j)}{\partial F^2(I_i)} \\ \frac{\partial\,\mathrm{ssd}}{\partial F^3(I_i)} &= \sum_{k\in N(j)} \frac{\partial\,\mathrm{ssd}}{\partial \phi_z(I_k)} \sum_{j\in N(i)} \frac{\partial \phi_z(I_k)}{\partial \phi_{z_h}(I_j)} \frac{\partial \phi_{z_h}(l_j)}{\partial F^3(I_i)} \end{aligned} \tag{3.50}$$

All the rest frameworks are same as the basic scheme. In this version, the computation time can be slightly improved. Fig. 3.3 shows the flowchart of the simplification scheme. In chapter 5, experiments are designed to compare the performance of the basic scheme and the simplified scheme.

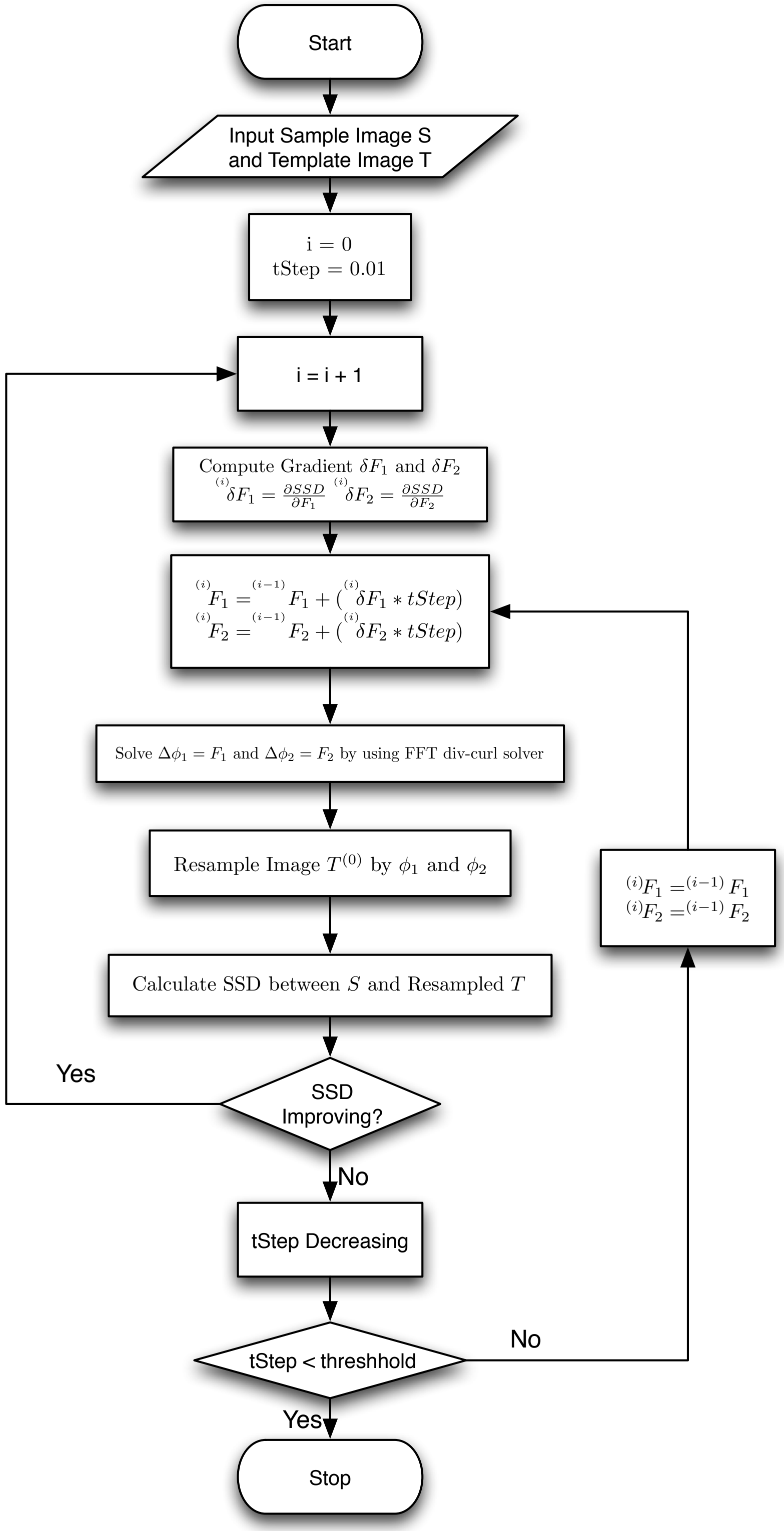


Figure 3.3. A flowchart of the simplification scheme of the proposed method.

3.4 An Improved Scheme

In the basic scheme, the required regularization is implicitly specified by the underlying div-curl system which may not be strong enough to warrant a folding free deformation field. To further strengthen the regularization, we can set the parameters as the divergence and curl of the B-spline coefficients of the displacement field, rather than the divergence and curl of the displacement field directly. This implies further smoothness of the deformation field. In practice, this is equivalent to the use of B-spline interpolation kernel of H in (3.46). The modification leads to the following set of equations.

$$\begin{aligned} \nabla \cdot C &= \frac{\partial C_x}{\partial x} + \frac{\partial C_y}{\partial y} \\ \nabla \times C &= \frac{\partial C_y}{\partial x} - \frac{\partial C_x}{\partial y} \end{aligned} \tag{3.51}$$

where C is the B-spline coefficient, and the relationship between C and ϕ is shown below,

$$\phi(\mathbf{x}) = \mathbf{x} + \sum_{\mathbf{i} \in I_c} c_{\mathbf{i}} \beta_3 (\frac{\mathbf{x}}{h} - \mathbf{i}) \tag{3.52}$$

Instead of solving for displacements ϕ, we now need to solve the B-spline coefficients. The displacements can be reconstructed using indirect B-spline transform [11]. B-spline transform has two operations - direct and indirect. Direct operation is to determine the B-spline coefficients for an equally spaced discrete signal. Indirect operation is a reverse operation which can reconstruct a signal from its B-spline coefficients with an optional zooming factor h (knot spacing). Assume a discrete signal $g(k), \forall k \in Z$ is given. The direct B-spline transformation and the indirect B-spline transform are used to find its B-spline coefficients and reconstruction respectively as shown in Fig 3.4. In Fig 3.4, $B_h^{n^{-1}}(z)$ represents the z-transform of the direct B-spline filter of order n, $B_h^n(z)$ represents the z-transform of the indirect B-spline filter of order n, h represents the knot spacing.

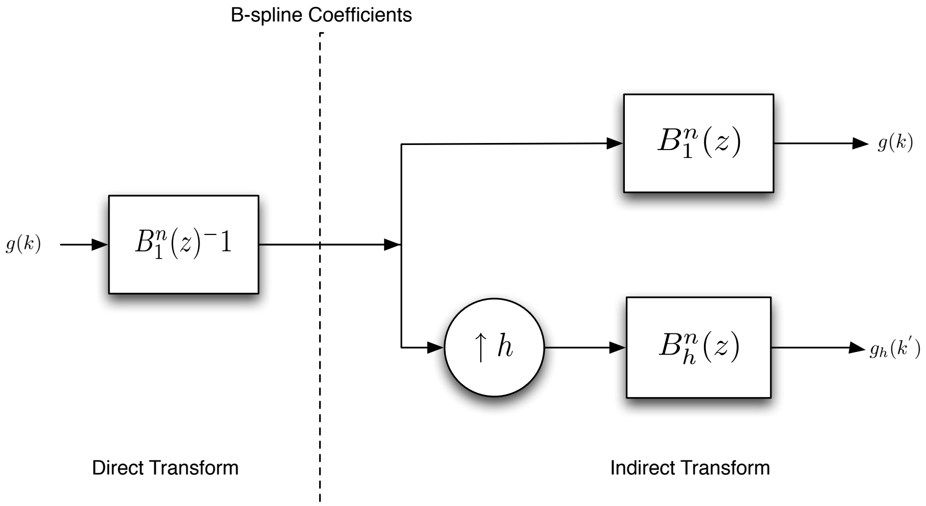
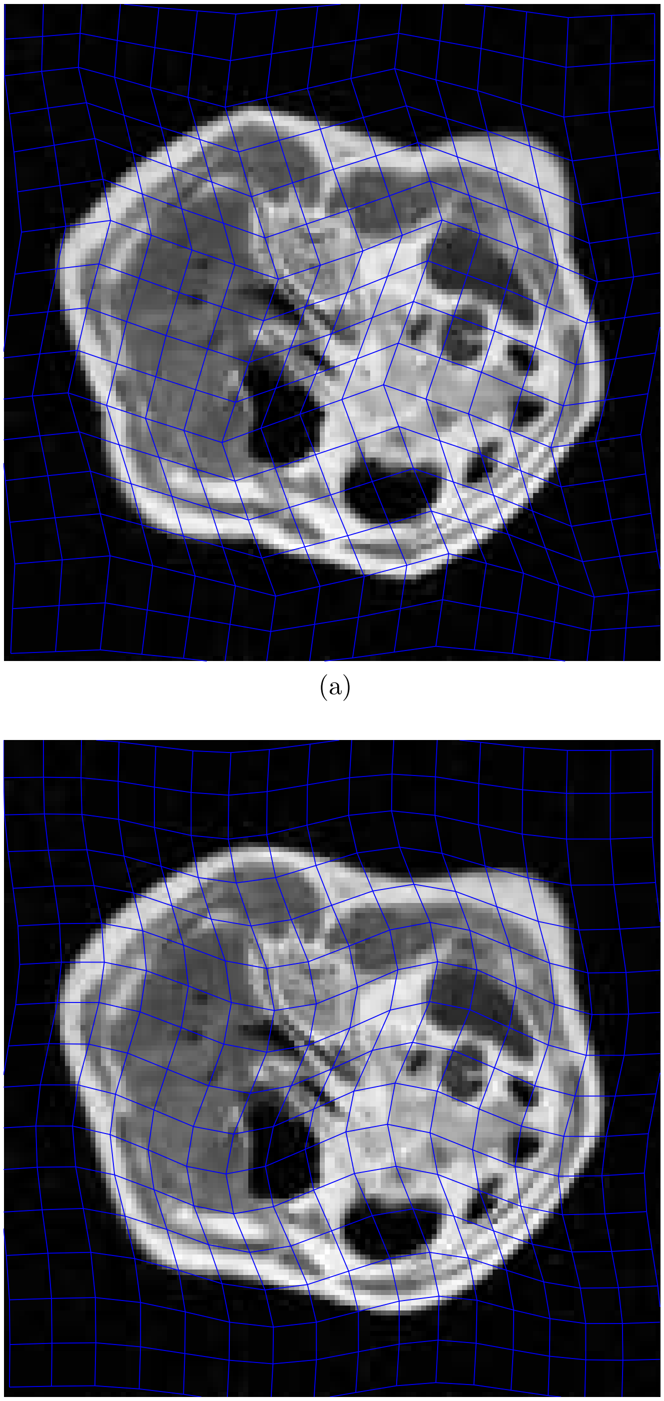


Figure 3.4. A diagram of a B-spline operation.

Figure 3.5 illustrates the difference between the results of the basic scheme and the improved scheme. It is obtained from our registration method which the goal is to register a squared 2-D MRI image (129 by 129 pixels) warped by thin-plate spline [21] back to its original form. In this case, the knot spacing is 16. In Fig. 3.5, the complete deformation field is obtained using linear interpolation from the deformation field defined at the coarse grid. Fig. 3.5 shows the results using B-spline interpolation. Clearly, the improved scheme results in a smoother deformation field than the basic scheme.



(a)

(b)

Figure 3.5. Grid generated by (a) basic scheme and (b) improved scheme.

Fig. 3.6 illustrating the idea of the proposed method with B-spline interpolation. The difference between this new approach and the original approach is shown in the double-line box of Fig. 3.6.

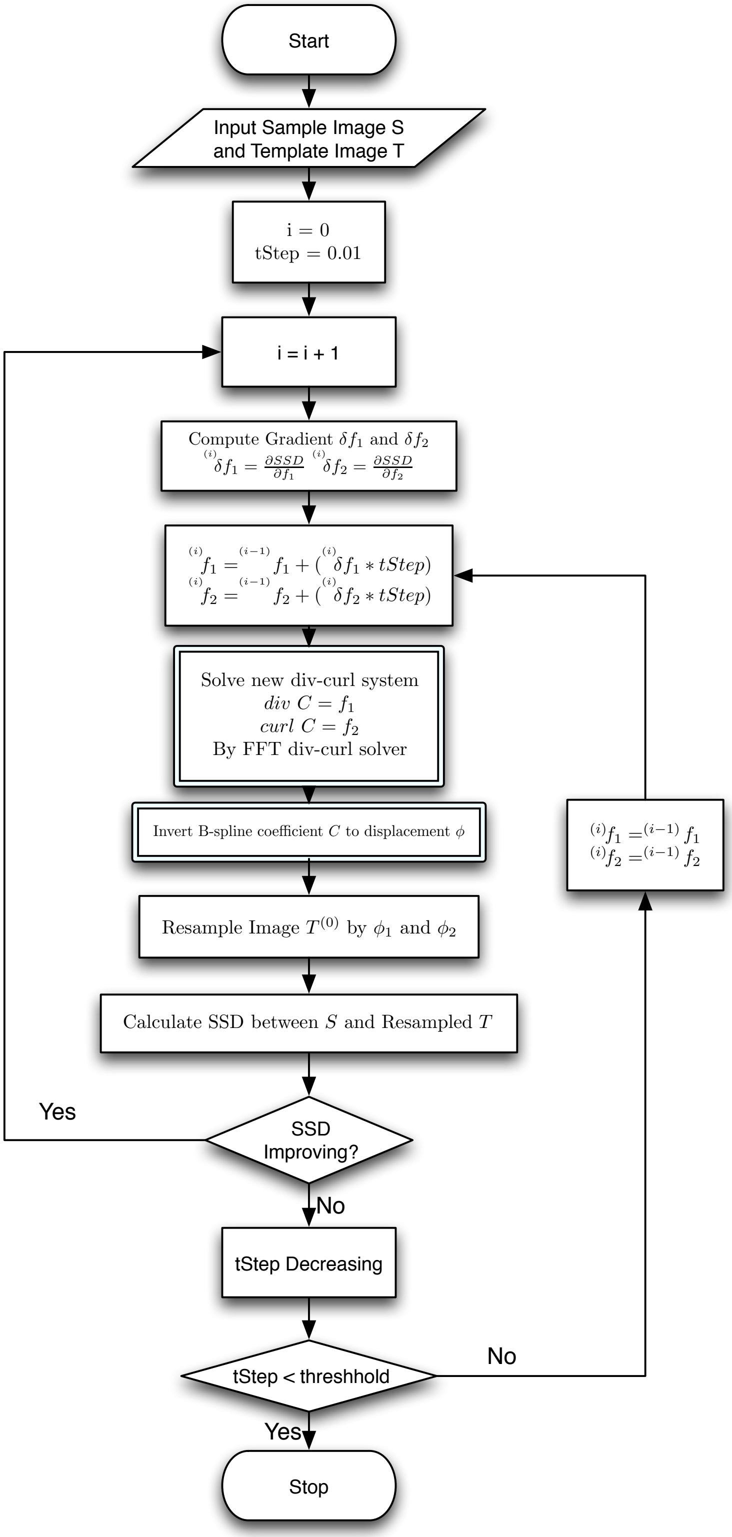


Figure 3.6. Original proposed method with B-spline interpolation.

3.5 A Interactive Scheme

All methods proposed in the previous sections provide robust result, however, it still has chance to register similar features which do not correspond physically. To avoid this situation, we borrow a idea from [10] and [22]. A second term $\mathbf{D}_L$ is added to the minimization formulation such that $\mathbf{ssd}_{total} = \mathbf{ssd} + \mathbf{D}_L$. The term $\mathbf{D}_L$ is calculated based on a set of landmark points which is supplied by the user, and is devised as,

$$\mathbf{D}_L = \sum_{i=1}^{d} \alpha_i \|\phi(\mathbf{x_i}) - \mathbf{z_i}\|^2 \tag{3.53}$$

where d is the number of the landmarks.

The gradient information of above method can be derived as following,

$$\begin{aligned}
\frac{\partial\,\mathrm{ssd}_{total}}{\partial f^J(I_i)} &= \frac{\partial\,\mathrm{ssd}}{\partial f^J(I_i)} + \frac{\partial\,\mathrm{D}_L}{\partial f^J(I_i)} \\
&= \sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial\phi_x(I_l)} \sum_{k\in N(j)} \frac{\partial\phi_x(I_l)}{\partial\phi_{x_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{x_h}(I_k)}{\partial F^1(I_j)} \frac{\partial F^1(I_j)}{\partial f^J(I_i)} \\
&+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial\phi_y(I_l)} \sum_{k\in N(j)} \frac{\partial\phi_y(I_l)}{\partial\phi_{y_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{y_h}(I_k)}{\partial F^2(I_j)} \frac{\partial F^2(I_j)}{\partial f^J(I_i)} \\
&+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{ssd}}{\partial\phi_z(I_l)} \sum_{k\in N(j)} \frac{\partial\phi^z(I_l)}{\partial\phi_{z_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{z_h}(I_k)}{\partial F^3(I_j)} \frac{\partial F^3(I_j)}{\partial f^J(I_i)} \\
&+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{D}_L}{\partial\phi_x(I_l)} \sum_{k\in N(j)} \frac{\partial\phi_x(I_l)}{\partial\phi_{x_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{x_h}(I_k)}{\partial F^1(I_j)} \frac{\partial F^1(I_j)}{\partial f^J(I_i)} \\
&+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{D}_L}{\partial\phi_y(I_l)} \sum_{k\in N(j)} \frac{\partial\phi_y(I_l)}{\partial\phi_{y_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{y_h}(I_k)}{\partial F^2(I_j)} \frac{\partial F^2(I_j)}{\partial f^J(I_i)} \\
&+ \sum_{l\in N(k)} \frac{\partial\,\mathrm{D}_L}{\partial\phi_z(I_l)} \sum_{k\in N(j)} \frac{\partial\phi_z(I_l)}{\partial\phi_{z_h}(I_k)} \sum_{j\in N(i)} \frac{\partial\phi_{z_h}(I_k)}{\partial F^3(I_j)} \frac{\partial F^3(I_j)}{\partial f^J(I_i)}
\end{aligned} \tag{3.54}$$

By the distributivity law of convolution, Eq. (3.54) can be simplified to

$$\begin{aligned}\frac{\partial\,\mathrm{ssd}_{total}}{\partial f^J(I_i)} = & \sum_{l\in N(k)}\left[\frac{\partial\,\mathrm{ssd}}{\partial\phi_x(I_l)}+\frac{\partial\,\mathrm{D}_L}{\partial\phi_x(I_l)}\right]\sum_{k\in N(j)}\frac{\partial\phi_x(I_l)}{\partial\phi_{x_h}(I_k)}\sum_{j\in N(i)}\frac{\partial\phi_{x_h}(I_k)}{\partial F^1(I_j)}\frac{\partial F^1(I_j)}{\partial f^J(I_i)}\\ & +\sum_{l\in N(k)}\left[\frac{\partial\,\mathrm{ssd}}{\partial\phi_y(I_l)}+\frac{\partial\,\mathrm{D}_L}{\partial\phi_y(I_l)}\right]\sum_{k\in N(j)}\frac{\partial\phi_y(I_l)}{\partial\phi_{y_h}(I_k)}\sum_{j\in N(i)}\frac{\partial\phi_{y_h}(I_k)}{\partial F^2(I_j)}\frac{\partial F^2(I_j)}{\partial f^J(I_i)}\\ & +\sum_{l\in N(k)}\left[\frac{\partial\,\mathrm{ssd}}{\partial\phi_z(I_l)}+\frac{\partial\,\mathrm{D}_L}{\partial\phi_z(I_l)}\right]\sum_{k\in N(j)}\frac{\partial\phi_z(I_l)}{\partial\phi_{z_h}(I_k)}\sum_{j\in N(i)}\frac{\partial\phi_{z_h}(I_k)}{\partial F^3(I_j)}\frac{\partial F^3(I_j)}{\partial f^J(I_i)}\end{aligned}\tag{3.55}$$

The explicit derivative of $\frac{\partial\,\mathrm{D}_L}{\partial\varphi_m}$ is

$$\frac{\partial\,\mathrm{D}_L}{\partial\phi_m} = 2\sum_{j=1}^{d}\alpha_j[\phi(\mathbf{x}_j)-\mathbf{z}_j]\tag{3.56}$$

where m represents the dimensional index.

In the next chapter, a set of experiments based on all the method proposed in this chapter will be presented.

CHAPTER 4

EXPERIMENTAL RESULTS

In this chapter, various experiments are presented and to evaluate the performance of our proposed system.

4.1 Data Sets

In this section, we describe three image sets and warping functions that will be used for the experiments.

4.1.1 An Artificial Image Pair

It is an artificial image. The size of the artificial image is 17-by-21 and the range of intensity value is $[0, 10]$. The template image T is shown in Fig. 4.1(a) and the reference image is shown in Fig. 4.1(b). In the reference image, the displacement of the left block is $(1, 1)$ and the displacement of the right block is $(-1, 1)$. This image set is used to test the aperture problem in 4.8.

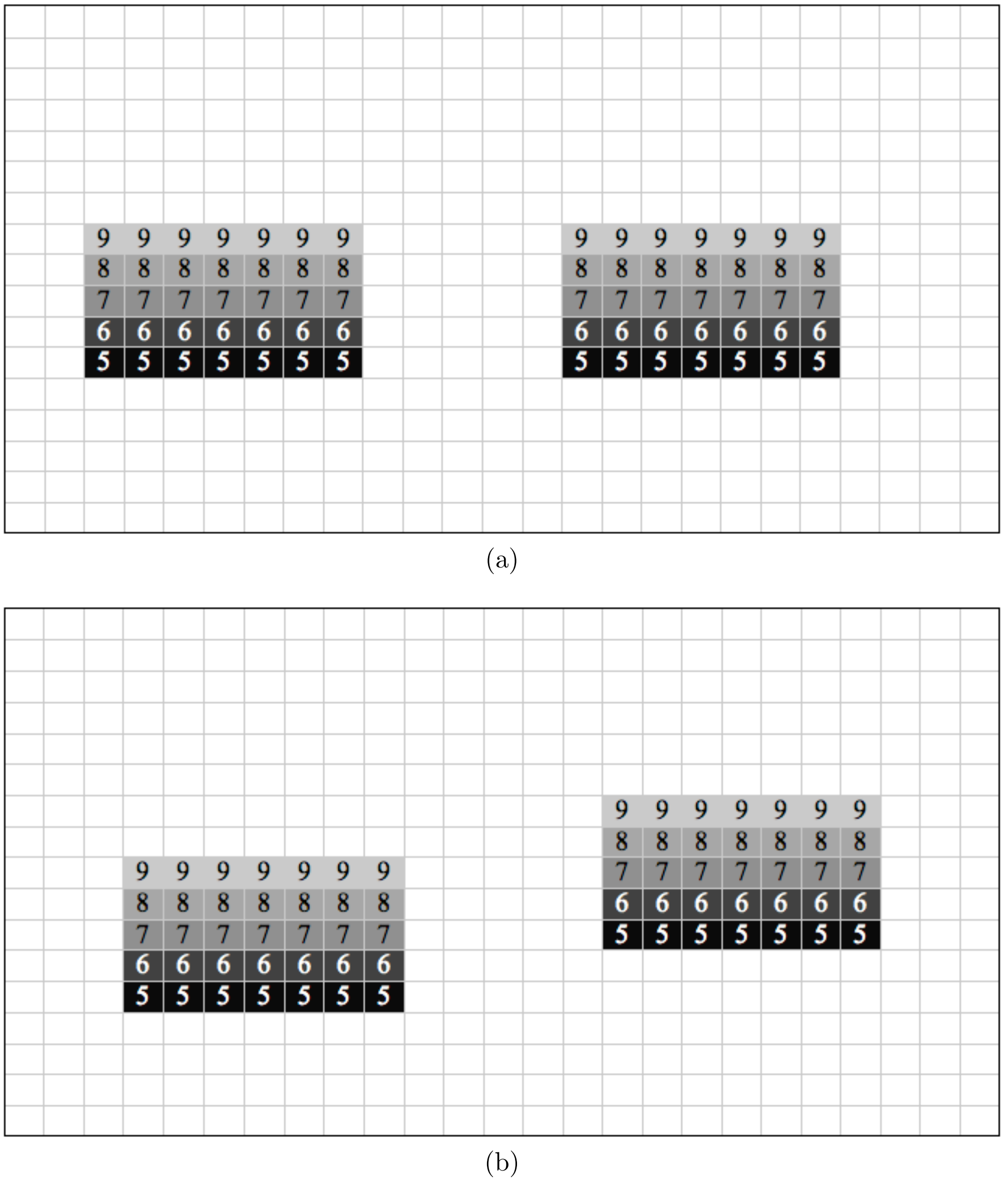


Figure 4.1. Artificial patterned image pair (a) Template Image T (b) Reference Image R.

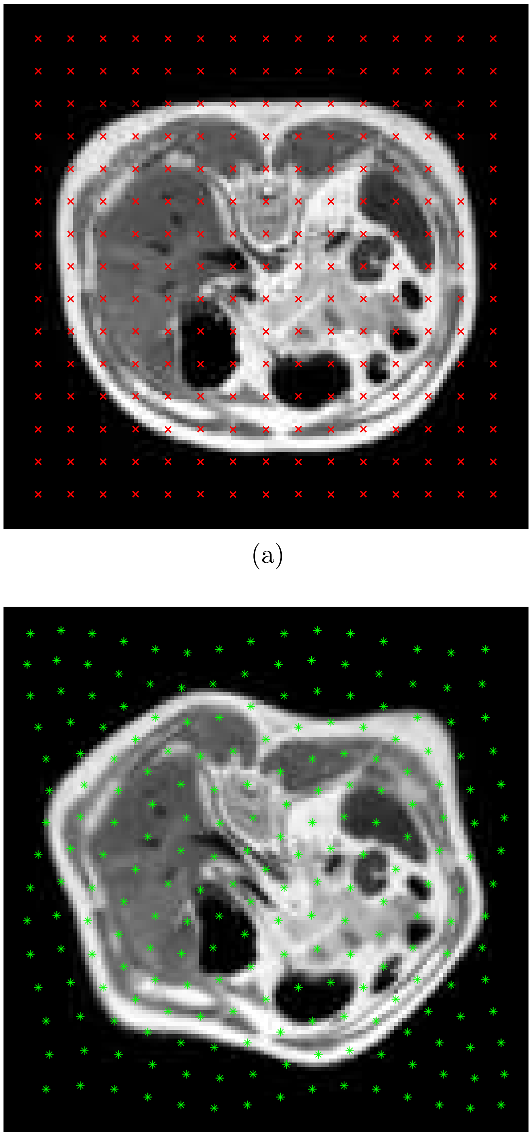
4.1.2 Clinical Medical Image with Synthetic Deformation Fields

In our experiments, we used a clinical medical image to test our system. The image is cropped from the red band of a slice of Visible Human male data. The original size of this image is 520-by-1024, which includes a human's celiac and his arms. We segmented the celiac part and resampled the image to 129-by-129 for multiresolution purpose. The intensity of the celiac image is $[0, 255]$. Since there is no true deformation came along with the image, we have to generate the true deformation artificially by using thin-plate spline warping method [21]. In thin-plate spline warping method, we need to provide two coordinate sets - original and destination. It will warp the original points to the destination points. We designed a 1D function to control the deformation level by thin-plate spline where T was warped based on 17-by-17 control points to form the reference image R (Fig. 4.2(b)). This 1D function is applied on both x and y axis. The positions of the control points are then altered in a deterministic manner controlled by a parameter d a ranging from 0 99 characterizing the degree of deformation. The original coordinate points are evenly distributed as shown in Fig. 4.2(a). Fig. 4.2(b) shows the destination coordinate points are driven by the following function with deformation degree 50.

$$U = 129 \cdot (\alpha_i)^{0.3} \cdot sin(4\pi\beta_i) \cdot (-d) \qquad (4.1)$$

$$1 \leq i \leq 17$$

where U is the displacement of correspond original coordinate points. α is a 1-by-17 scalar which has value from $0, 1, \ldots, 7, 8, 7, \ldots, 1, 0$. β is a 1-by-17 scalar which has value from 0 to 1 with interval $1/(17-1)$. d represents the deformation degree which is to control the deformation level, if d increases, the reference image will be deformed larger.



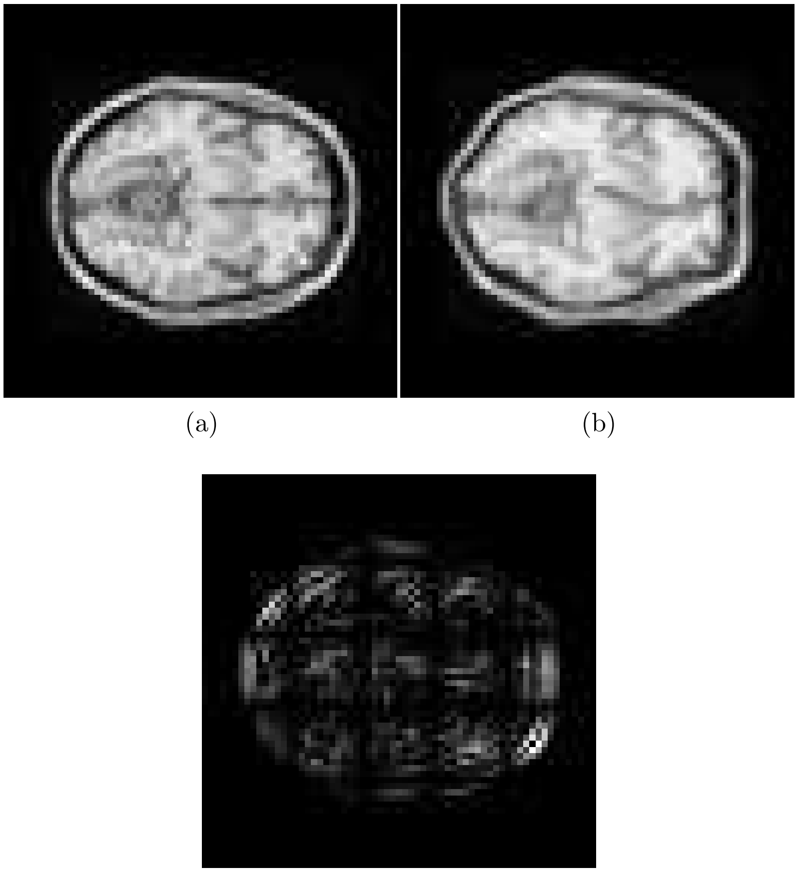
(a)

(b)

Figure 4.2. Clinical celiac MRI image pair with synthetic deformation field (a) Template Image T (b) Reference Image R.

4.1.3 3D Image Volume With Synthetic Deformation Fields

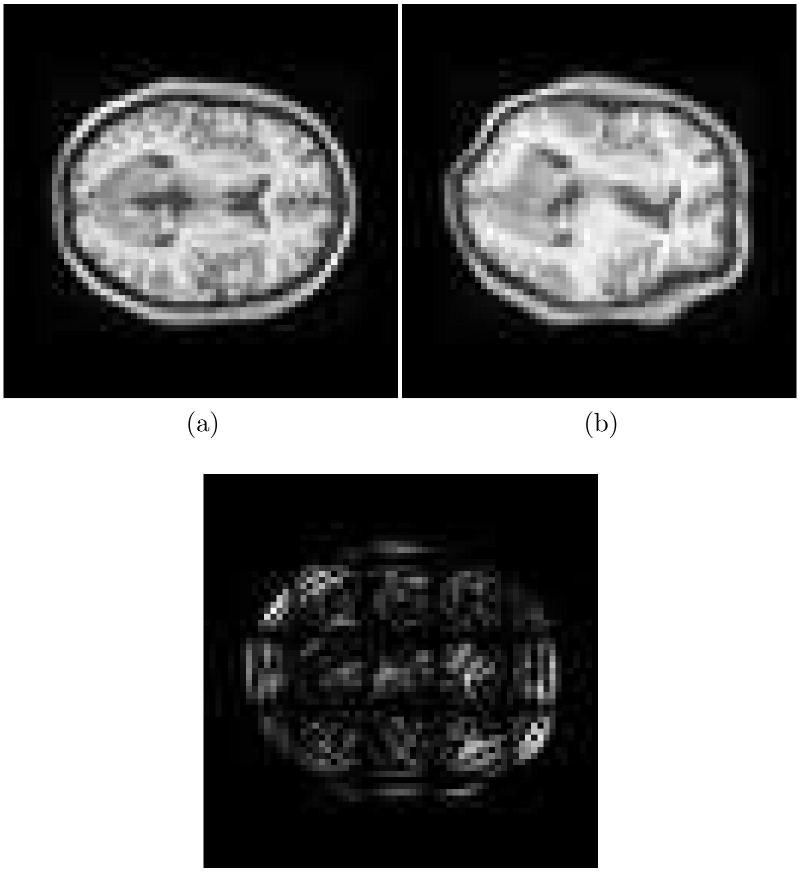
A 3D MRI brain image volume that downloaded from McConnell Brain Imaging Centre (BIC) of the Montreal Neurological Institute, McGill University is used to test our proposed method. The original image volume size was 217x181x181. To facilitate multiresolution optimization and speed up the registration process, the image volumes were resampled to produce 65x65x65 image volumes. The range of intensity is $[0, 255]$. Again, the reference image volume is warped by using thin-plate spline warping method [21] based on 7-by-7-by-7 grid points. Fig. 4.3, 4.4, and 4.5 shows the 17^{th}, 33^{rd}, and 45^{th} slice of the image volume respectively. The intensity of residue images in fig 4.3 through 4.5 are enhanced for display purpose.



(a) (b)

(c)

Figure 4.3. The 17^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.

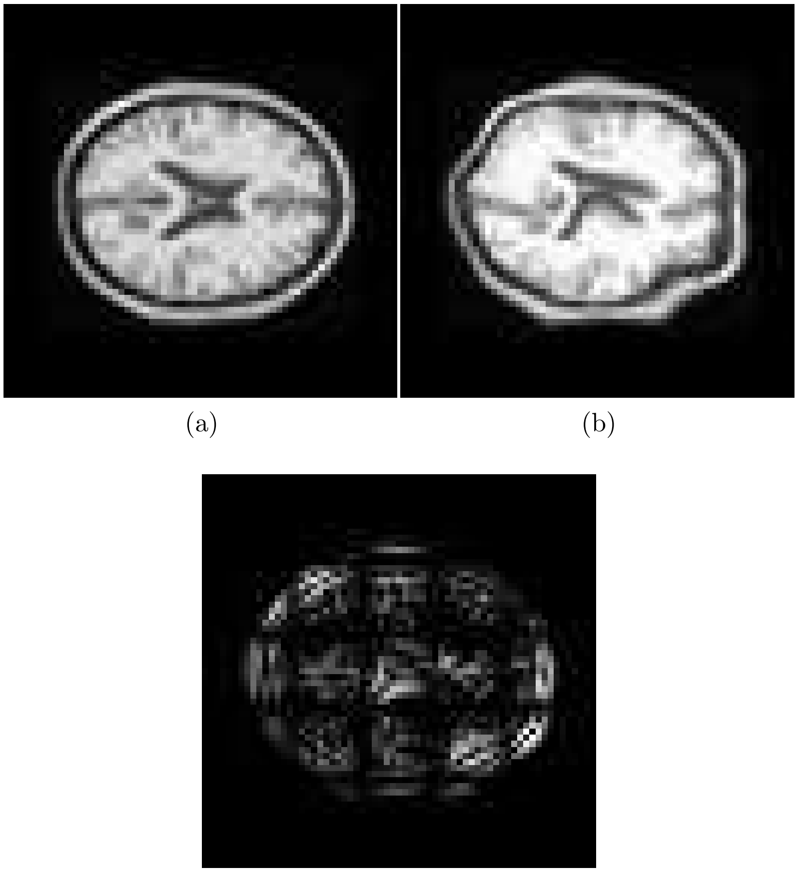


(a)

(b)

(c)

Figure 4.4. The 33^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.



(a) (b)

(c)

Figure 4.5. The 45^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.

4.2 Evaluation Criterion

In our experiments, several criterions are evaluated to ensure the credibility of the proposed methods.

Accuracy: Two criterions are measured in this category.

- Sum of squared differences (SSD): It is a most often and reasonable way to measure the discrepancy between two images.

$$SSD = \sum_{i \in I} (T(\varphi(i)) - R(i))^2 \tag{4.2}$$

- Warping index: Note the goal of the proposed methods is minimizing the SSD, however, lower SSD may appear because of the aperture problem which does not represent the correctness of solution. Therefore, we compared our solution with the true deformation.

$$w = \frac{1}{\|A\|} \sum_{i \in A} \|\phi(i) - \hat{\phi}(i)\| \tag{4.3}$$

where $A \subseteq R$ is the region of interest. $\hat{\phi}$ represents the true deformation

Jacobian determinant: Grid folding problem is prohibited. It represents the solution falls to a incorrect area. The Jacobian determinant can provide a important information when making a change of variables when integrating a function over its domain. The minimum of the Jacobian determinant of the estimated deformation field is monitored, which is utilized as an indication whether grid folding is resulted or not.

Robustness: This criterion is to measure how large deformation that the propose can handle. Eq. (4.1.2) is used to generate the deformed image and its true deformation. We iterated the experiments by modified the deformation degree d. If a experiment could not pass its accuracy measurement, the robustness of that method is its deformation degree d.

Efficiency: This criterion is to measure the runtime and the convergence rate of the proposed methods. The runtime is measured in seconds and the Converge rate is measured in number of iterations.

4.3 Optimization and Stop Criterion

As described in 3.2.3, the gradient descent optimization is employed in the proposed method. The time step is assigned to whole divergence and curl value simultaneously and will be increased if the SSD improved, and decreased otherwise. The stop criterion is set when the time step is lower the 10^{-7}.

4.4 Experiment 1: Find the Optimal Size of Inverse Filter

- Purpose: In 3.2.2.3, we have mentioned the inverse filter is an approximative solution in our case. Thus the optimal size of inverse filter cannot be determined mathematically. The purpose of this experiment is to determined if a optimal inverse filter existed.
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made based on its accuracy, Jacobian determinant, robustness, and efficiency.
- Method Used: Basic scheme.
- Experiment: The various filter sizes were tested [3x3, 1/4 of image size, 1/2 of image size, same as image size, double of image size]. The deformation degree d is changed from 5 to 50 with interval 5.

Table 4.1 shows the original value of SSD, warping index, and Jacobian determinant of each deformation degree.

Table 4.1. Original values of SSD, warping index, and Jacobian determinant of each deformation degree

Deformation Degree	Original SSD	Mean Warping	Max Warping	Min Jacobian
5	74.40	0.42	0.66	0.96
10	275.47	0.84	1.33	0.92
15	534.20	1.27	1.99	0.88
20	831.57	1.70	2.66	0.85
25	1174.30	2.13	3.32	0.81
30	1521.17	2.55	3.98	0.77
35	1862.73	2.98	4.65	0.73
40	2219.33	3.40	5.31	0.69
45	2545.20	3.83	5.97	0.65
50	2872.83	4.26	6.64	0.61

4.4.1 Results

Table 4.2. Results of inverse filter test

	Robustness	Accuracy				Efficiency	
Filter Size	Deformation Degree	SSD	Mean Warping	Max Warping	Min J	Iterations	Runtime (sec)
3x3	5	2.69	0.42	1.51	-0.21	180	31.15
33x33	50	1.22	0.20	1.35	0.31	1104	179.21
65x65	50	0.47	0.10	0.57	0.71	1694	450.15
129x129	50	2.92	0.19	0.92	0.62	2000	859.73
259x259	50	17.99	0.42	1.65	0.64	1929	1801.79

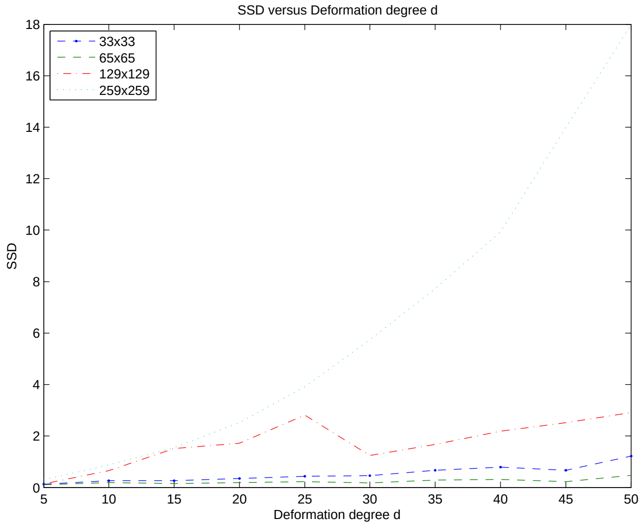


Figure 4.6. Results of experiment 1: SSD versus Deformation degree d.

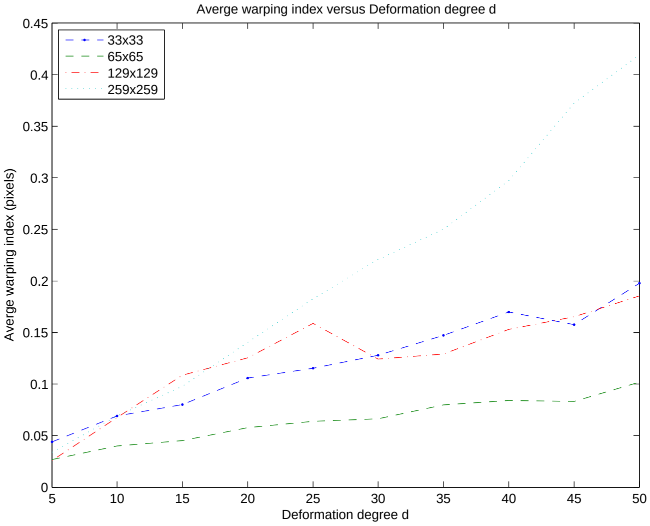


Figure 4.7. Results of experiment 1: Average warping index versus Deformation degree *d*.

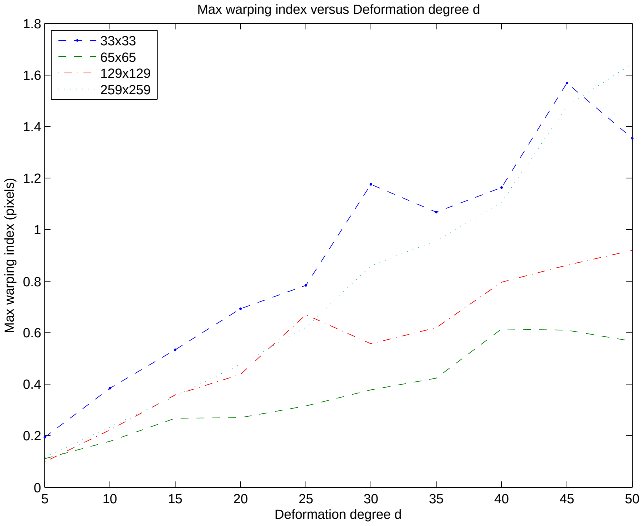


Figure 4.8. Results of experiment 1: Maximum warping index versus Deformation degree *d*.

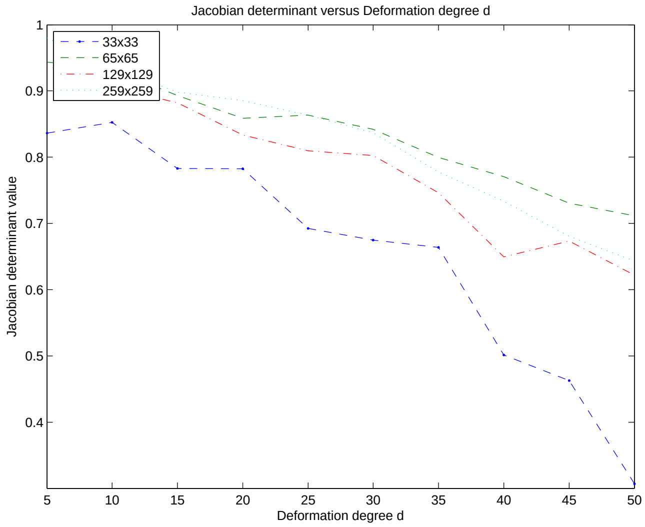


Figure 4.9. Results of experiment 1: Minimum Jacobian value versus Deformation degree d.

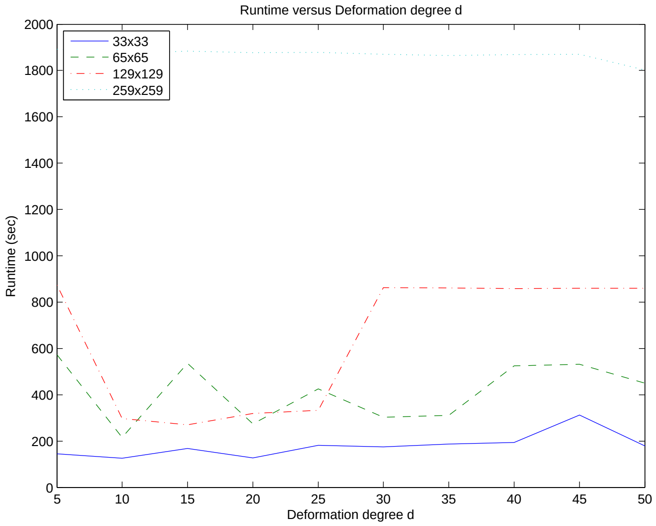


Figure 4.10. Results of experiment 1: Runtime versus Deformation degree d.

4.4.2 Discussion

In table 4.2, we observed the inverse filter with size 3x3 failed the robustness test at the very early stage, because of the Jacobian determinant is less than zero which means grid folding. From fig. 4.6, we observed the inverse filter with 1/4 and half of the image size generated better value in SSD. However, from the warping index figure (4.7 and 4.8), the inverse filter with 1/4 and double of image size did not generate good result as others. In 3.2.2.3, we have mentioned the inverse filter is an approximative solution in our case. As we expected, it does not generate a very consistent solution. It all depends on what kind of image you are testing. From this experiment, we chose the inverse filter with half of image size is the best solution. In the next section, a comparison between the inverse filter and other div-curl solvers is provided.

4.5 Experiment 2: Choose A Optimal Div-Curl Solver

- Purpose: In 3.2.2, we have introduced four div-curl solvers. This experiment is to help us to determine a div-curl solver which can bring us the best solution of the proposed methods.
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made based on its accuracy, robustness, Jacobian determinant and efficiency.
- Method Used: Basic scheme.
- Experiment: Due to the limitation of the direct method, it cannot be tested on a image with size 129-by-129. Therefore, the direct method is excluded from this experiment. We also set a the stop criterion for SOR to speed up the process, thus, SOR is considered an approximation also. Each SOR process is

terminated at the 100^{th} iteration. The deformation degree d is changed from 50 to 100 with interval 5.

4.5.1 Results

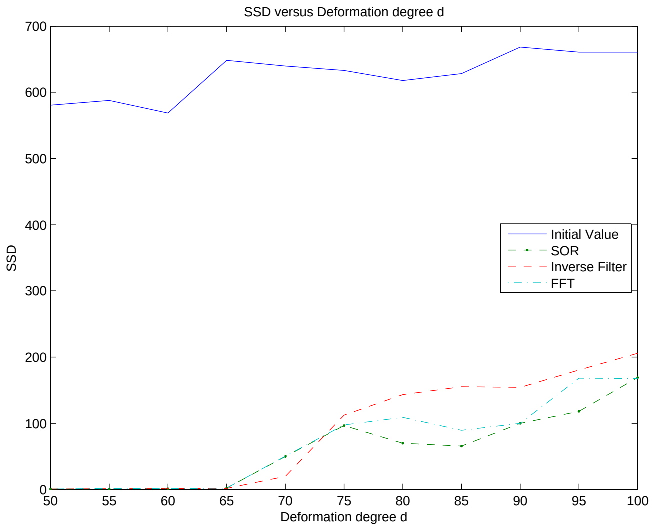


Figure 4.11. Results of experiment 2: SSD versus Deformation degree d.

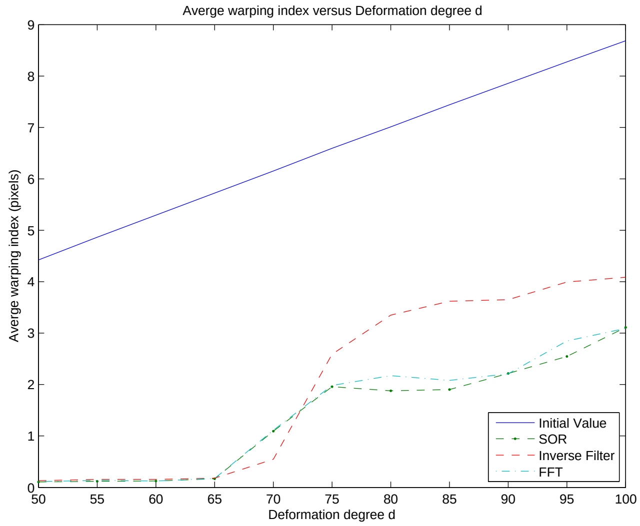


Figure 4.12. Results of experiment 2: Average warping index versus Deformation degree *d*.

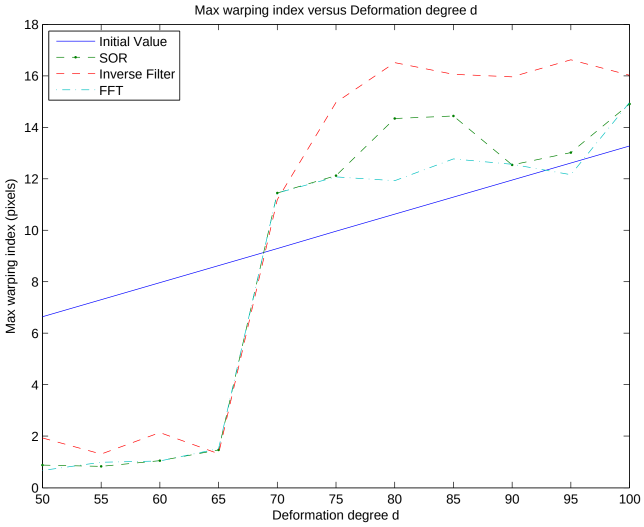


Figure 4.13. Results of experiment 2: Maximum warping index versus Deformation degree *d*.

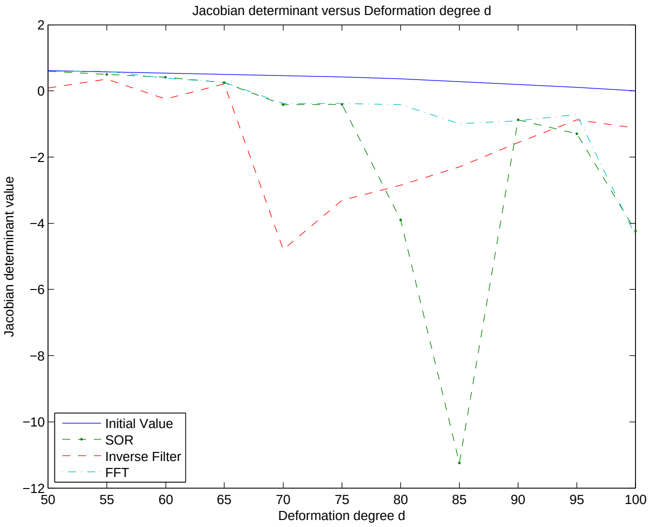


Figure 4.14. Results of experiment 2: Minimum Jacobian value versus Deformation degree d.

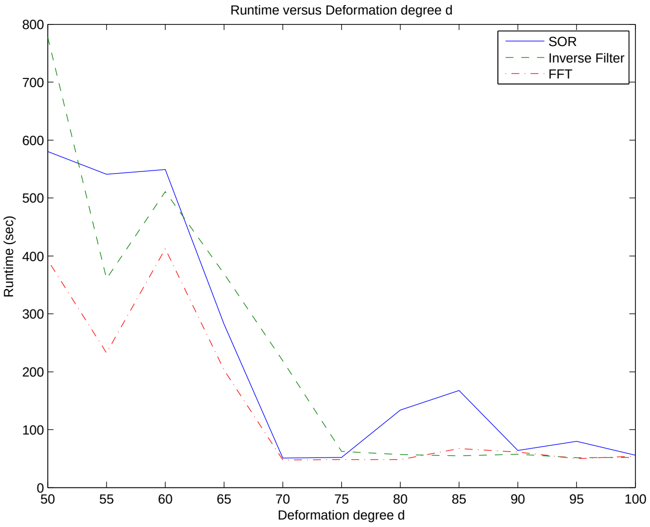


Figure 4.15. Results of experiment 2: Runtime versus Deformation degree d.

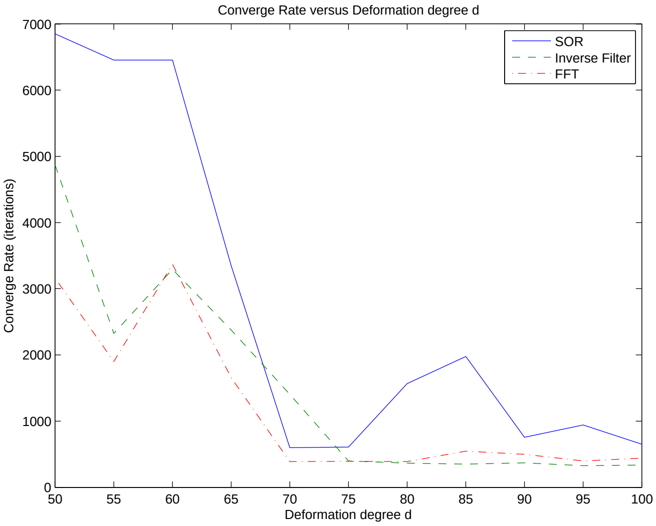


Figure 4.16. Results of experiment 2: Converge rate versus Deformation degree *d*.

4.5.2 Discussion

Fig. 4.11 shows all three div-curl solvers generated similar trend of SSD. From warping index (Fig. 4.12 and 4.13), we observed all three div-curl solver failed the robustness test after deformation degree $d = 65$ because it leads to wrong solutions (value of max warping index getting higher). Also, from the warping index figures, the SOR and FFT-based div-curl solver generated the results that a bit better than inverse filter div-curl solver. Although the results generated by the inverse filter div-curl solver are acceptable, however, it has to be excluded from our choices because of the size of the inverse filter cannot be determined consistently. It does not conclude the inverse filter based div-curl solver is useless, it still can handle the smaller deformed image with smaller size. The results generated by SOR, and FFT-based div-curl solver are almost exactly the same and the only difference is its complexity. In this experiment, we manually force the SOR stopped at the 100^{th} iteration to speed up the runtime (Fig. 4.15). The results generated by SOR are still acceptable and the runtime is improved apparently, however, FFT-based div-curl solver still has better performance. Therefore, FFT-based div-curl solver is chosen for the proposed method.

4.6 Experiment 3: Effects of Multiresolution Technique

- Purpose: Multiresolution technique is to speed up the registration process and increase its robustness by approaching the solution with gradual refinement. In this experiment, we tested the proposed method with and without multiresolution technique.
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made based on its accuracy, robustness, and efficiency.

- Method Used: Basic scheme.
- Experiment: The experiment compared the basic scheme of the proposed methods with multiresolution which started from knot spacing $h = 16$ and down to $h = 1$ and without multiresolution ($h = 1$). The deformation degree d is changed from 50 to 100 with interval 5.

4.6.1 Results

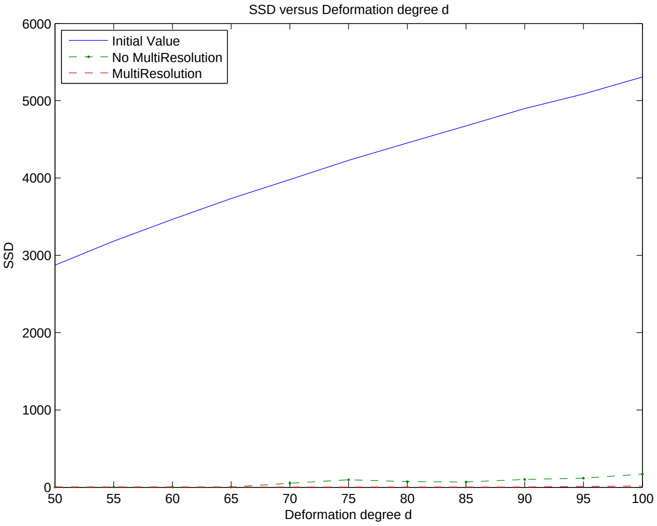


Figure 4.17. Results of experiment 3: SSD versus Deformation degree d.

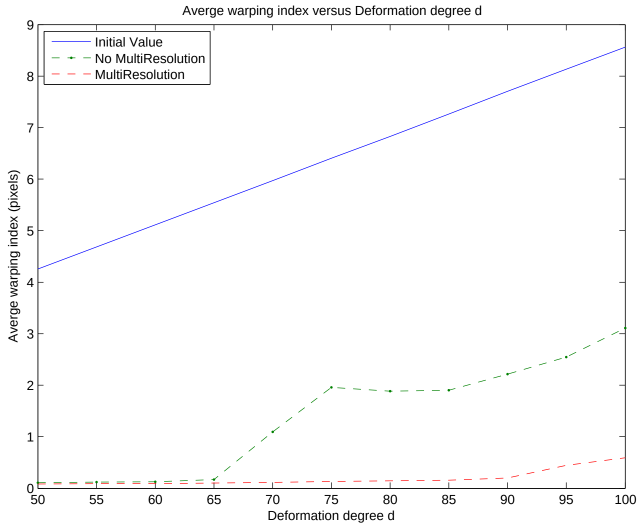


Figure 4.18. Results of experiment 3: Average warping index versus Deformation degree *d*.

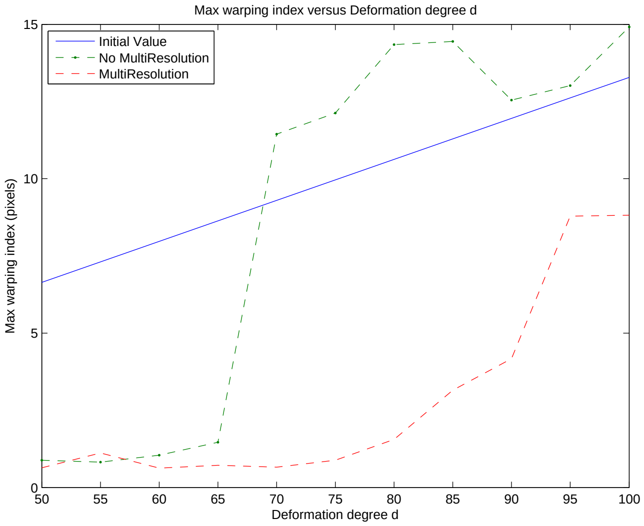


Figure 4.19. Results of experiment 3: Maximum warping index versus Deformation degree *d*.

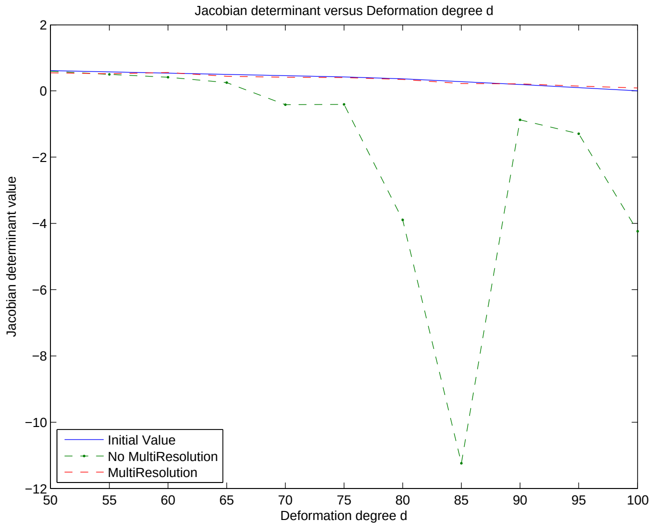


Figure 4.20. Results of experiment 3: Minimum Jacobian value versus Deformation degree d.

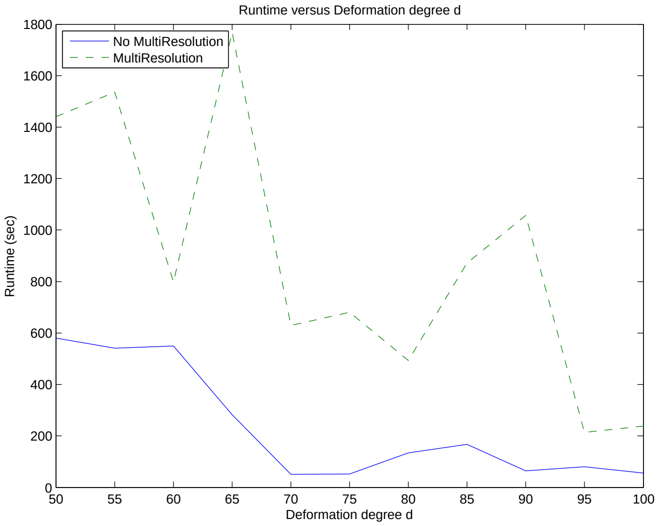


Figure 4.21. Results of experiment 3: Runtime versus Deformation degree d.

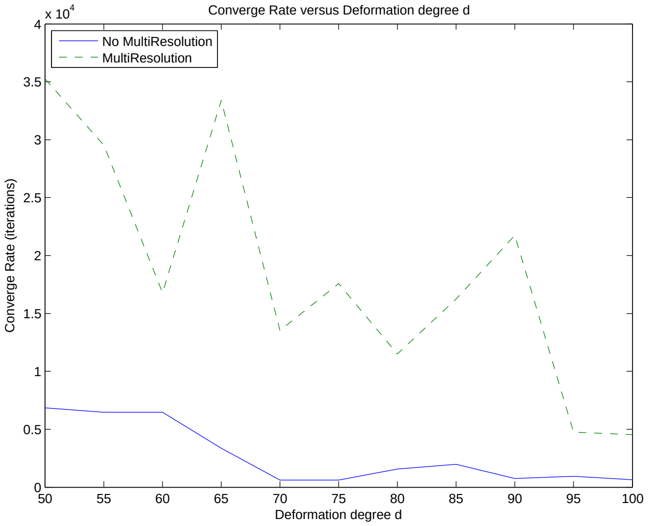


Figure 4.22. Results of experiment 3: Converge rate versus Deformation degree d.

4.6.2 Discussion

It is very obvious that the proposed method with multiresolution technique is better. The results generated by proposed method without multiresolution technique failed the warping index test and the max warping index test after the deformation degree $d = 65$ while the proposed method with multiresolution technique failed after $d = 85$. During the experiment, we also observed the results generated at $h = 1$ are worse than $h = 4$. It indicates a control grid is too fine will overcompensate for true image differences and noise. The spirit of multiresolution technique is to speed up the process. However, in our experiment, the convergence speed of multiresolution is not faster than single resolution, which is contrary of the multiresolution technique. The reason is because the increase/decrease rate of the time step is set smaller. If we set it larger, the convergence will improve greatly. However, to keep the experiments consistently, we used the same criterion for all experiments.

4.7 Experiment 4: Performance of Proposed Methods

- Purpose: We have presented the Helmholtz's theorem based parametric non-rigid image registration 3.2.1 and its simplified 3.3 and improved version 3.4. In this experiment, we would like to find out which version is the best method.
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made based on its accuracy, robustness, Jacobian determinant and efficiency.
- Method Used: Basic scheme, simplified scheme, and improved scheme with FFT-based div-curl solver and multiresolution technique embedded.
- Experiment: Three schemes run separately with the deformation degree d is changed from 50 to 100 with interval 5.

4.7.1 Results

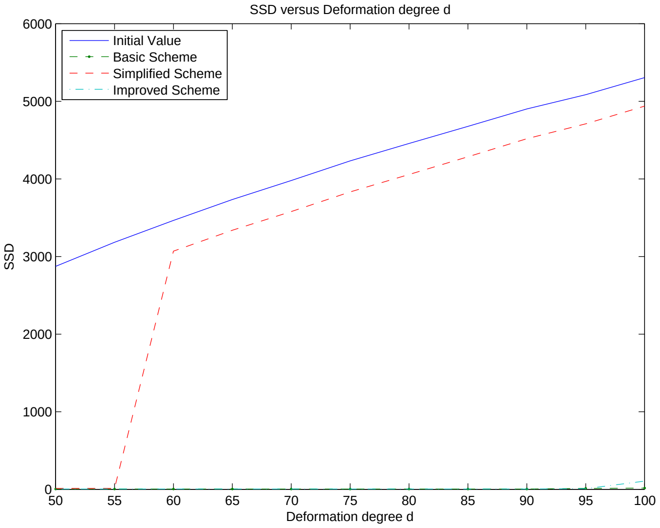


Figure 4.23. Results of experiment 4: SSD versus Deformation degree *d*.

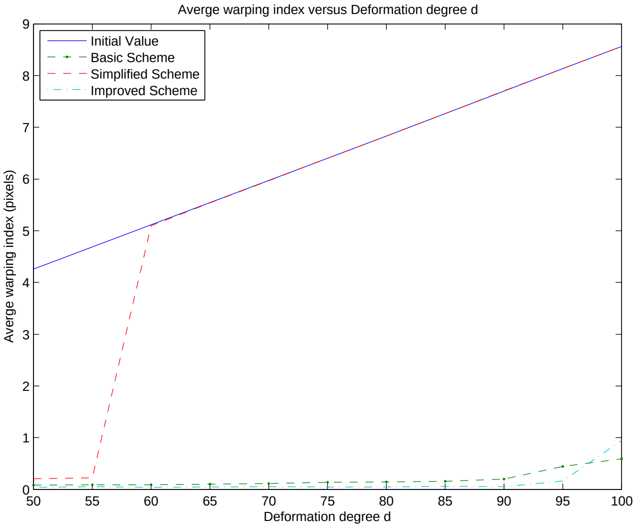


Figure 4.24. Results of experiment 4: Average warping index versus Deformation degree *d*.

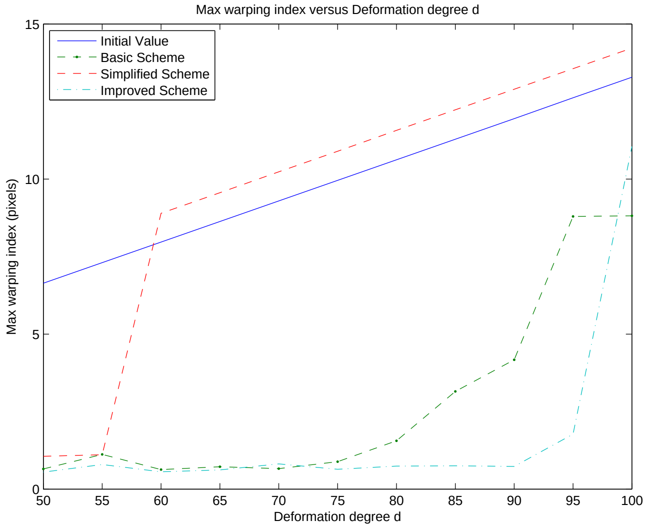


Figure 4.25. Results of experiment 4: Maximum warping index versus Deformation degree *d*.

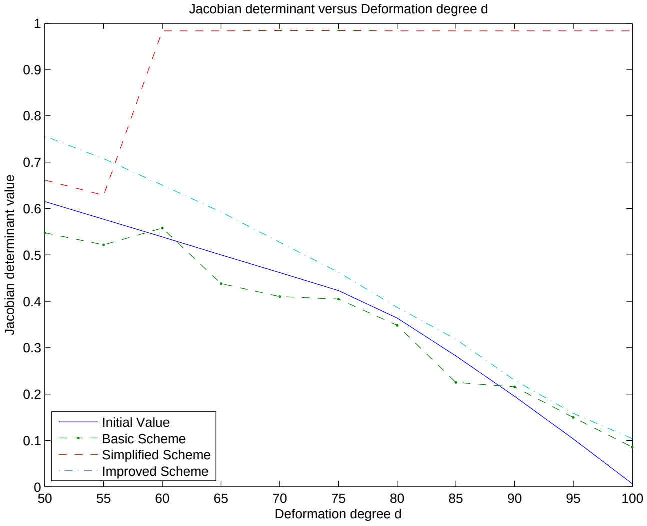


Figure 4.26. Results of experiment 4: Minimum Jacobian value versus Deformation degree *d*.

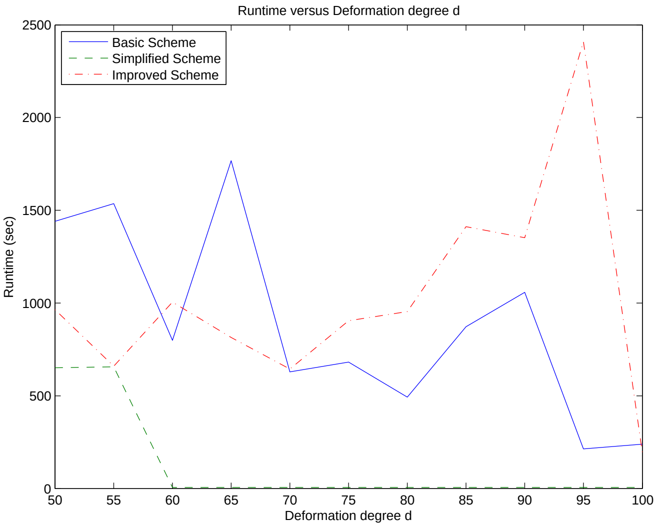


Figure 4.27. Results of experiment 4: Runtime versus Deformation degree d.

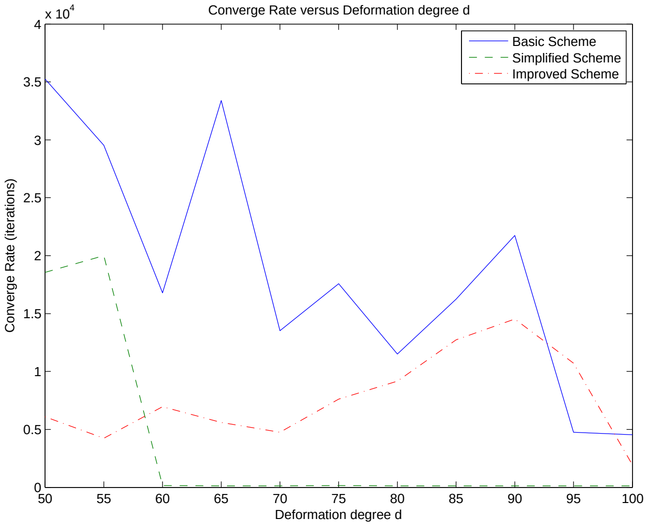


Figure 4.28. Results of experiment 4: Converge rate versus Deformation degree d.

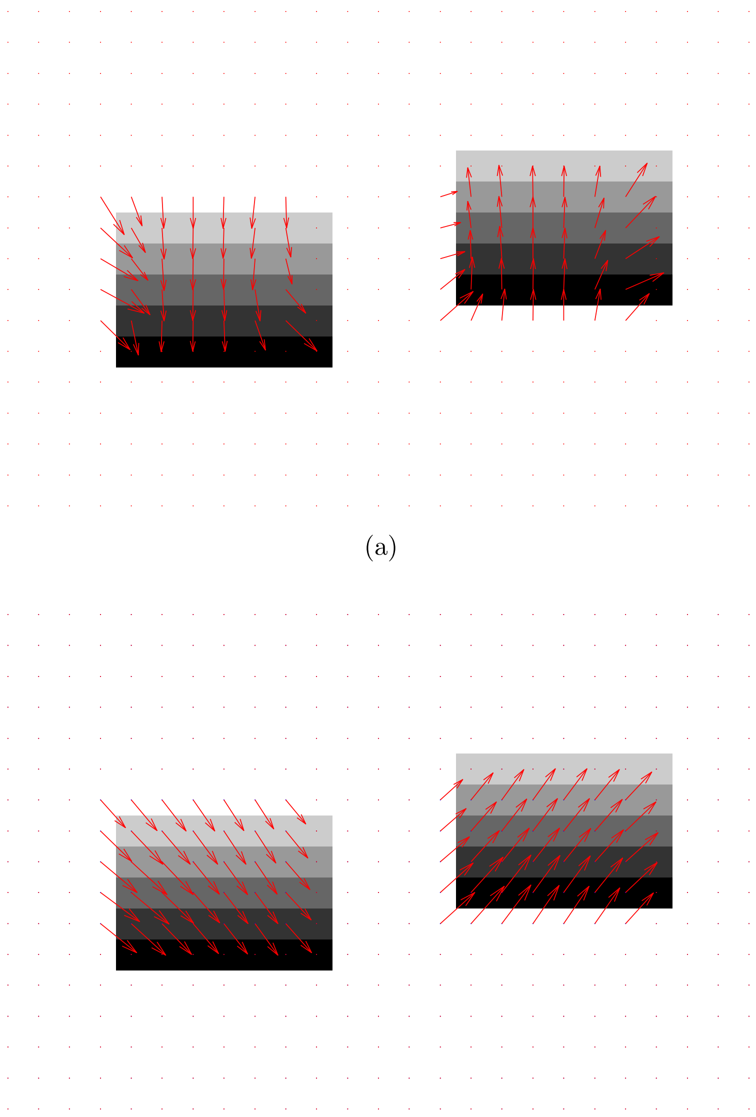
4.7.2 Discussion

We realized the complexity of the simplification scheme of proposed method is lower than others. However, the performance is not well as others. From the accuracy test, the simplification scheme failed at the deformation degree $d = 55$. It indirectly proves the Poisson equation that composed by a div-curl system can provides more accurate results. As our assumption in Chap. 3.4, the regularization is further strengthened by the B-spline interpolation. Fig. 4.26 shows the proposed method with B-spline interpolation has less chance to occur the grid folding. Also, from the warping index test (Fig. 4.24 and 4.25), it shows the improved scheme can handle more larger deformed images than other proposed methods.

4.8 Experiment 5: Ability of Surmounting The Aperture Problem

- Purpose: The aperture problem addressed a motion of a homogeneous contour is locally ambiguous, i.e., within a aperture, different physical motions are indistinguishable. This experiment is to show the ability of handling the aperture problem of the proposed method.
- Data Set Used: Image sets in 4.1.1
- Evaluation Criterion: The decision is made based on its accuracy of the warping index.
- Method Used: Improved scheme with FFT-based div-curl solver embedded, and the fast parametric elastic image registration.
- Experiment: We compared the proposed method with the fast parametric elastic image registration [10] which was found it suffer the aperture problem previously [23].

4.8.1 Results



(a)

(b)

Figure 4.29. Deformation field generated by (a) Fast parametric elastic image registration (b)Proposed method.

4.8.2 Discussion

The resulting deformation field is shown in Fig. 4.29(a). Clearly, the displacements from in the direction perpendicular to the direction of the image gradient can not be recovered. This is a common problem encountered in most gradient based non-rigid image registration techniques. The result obtained using the proposed method is shown in Fig. 4.29(b). It clearly indicates its resistance to this aperture problem. The reason for this is that by updating each divergence and curl value, it has a global effect on the deformation. Therefore, each time divergence and curl values on a grid point are updated, whole deformation field is altered. When the divergence and curl values adjusts itself to optimize the similarity measure by matching the discriminent information between two images, the deformation field on the part of the image lacking discriminant information is also determined.

4.9 Experiment 6: Proposed Method V.S. Fast Parametric Elastic Image Registration Method

- Purpose: From the above experiments, we have found a optimal combination of the proposed methods. In this experiment, we would like to compare the selected method to a existing parametric based non-rigid image registration - Fast parametric elastic image registration [10].
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made accuracy, robustness, Jacobian determinant and efficiency.
- Method Used: Improved scheme with FFT-based div-curl solver and multiresolution embedded, and the fast parametric elastic image registration.
- Experiment: We compared the proposed method with the fast parametric elastic image registration [10]. Multiresolution technique is applied on both methods

$(h = 16 \sim 4)$. The deformation degree d is changed from 50 to 100 with interval 1.

4.9.1 Results

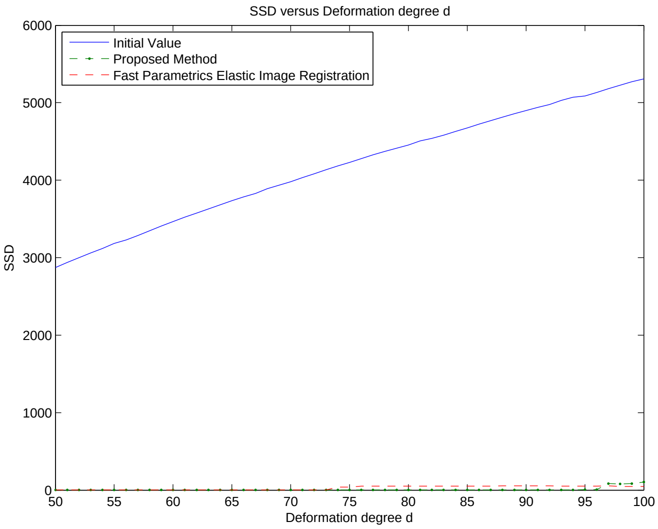


Figure 4.30. Results of experiment 6: SSD versus Deformation degree d.

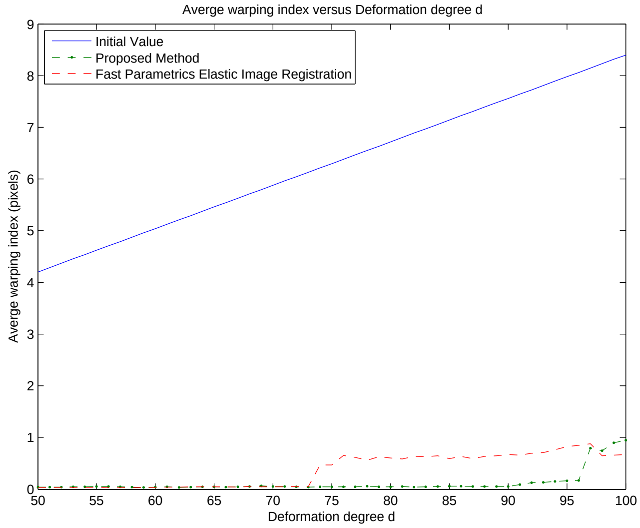


Figure 4.31. Results of experiment 6: Average warping index versus Deformation degree *d*.

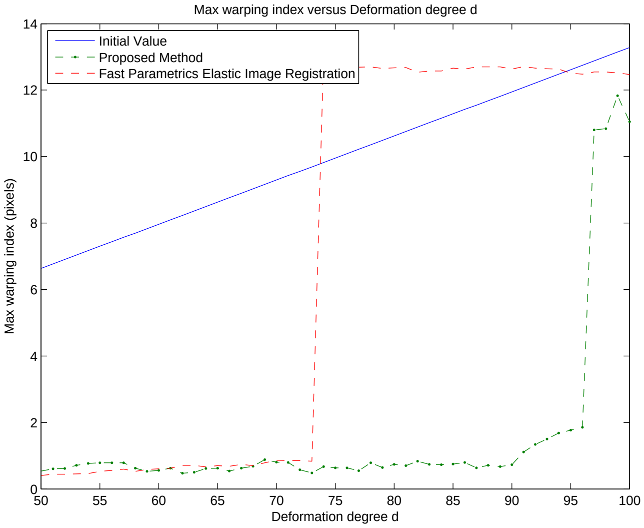


Figure 4.32. Results of experiment 6: Maximum warping index versus Deformation degree *d*.

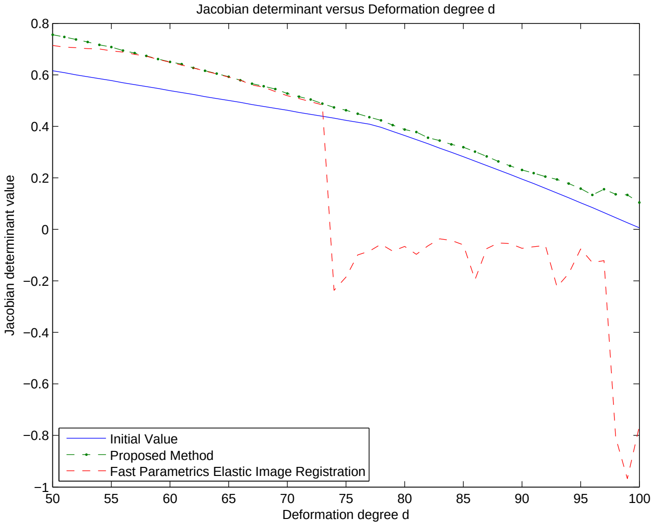


Figure 4.33. Results of experiment 6: Minimum Jacobian value versus Deformation degree *d*.

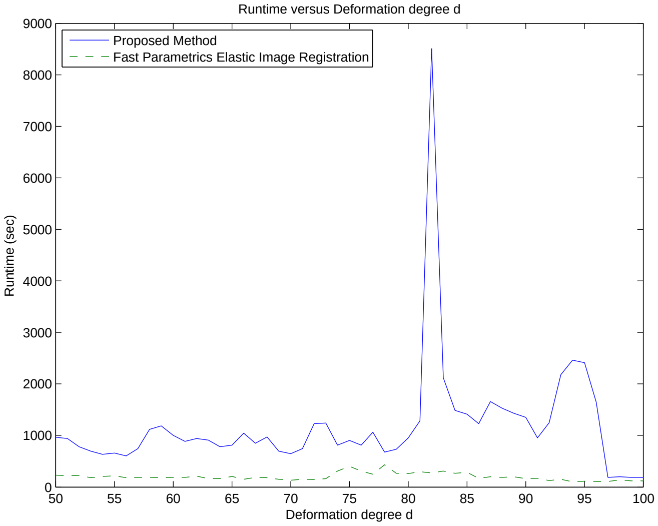


Figure 4.34. Results of experiment 6: Runtime versus Deformation degree *d*.

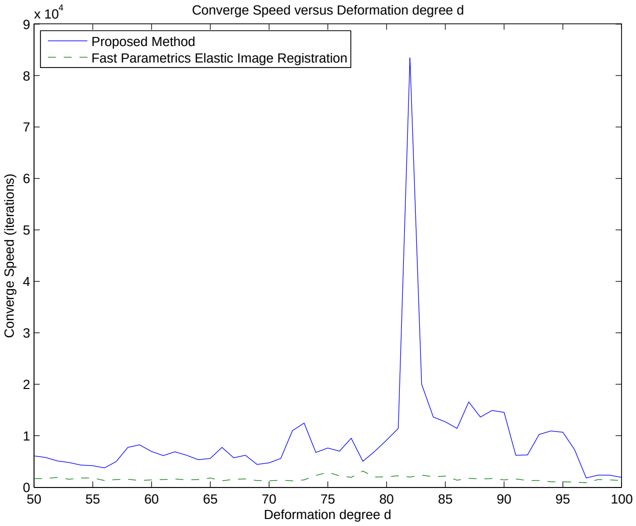


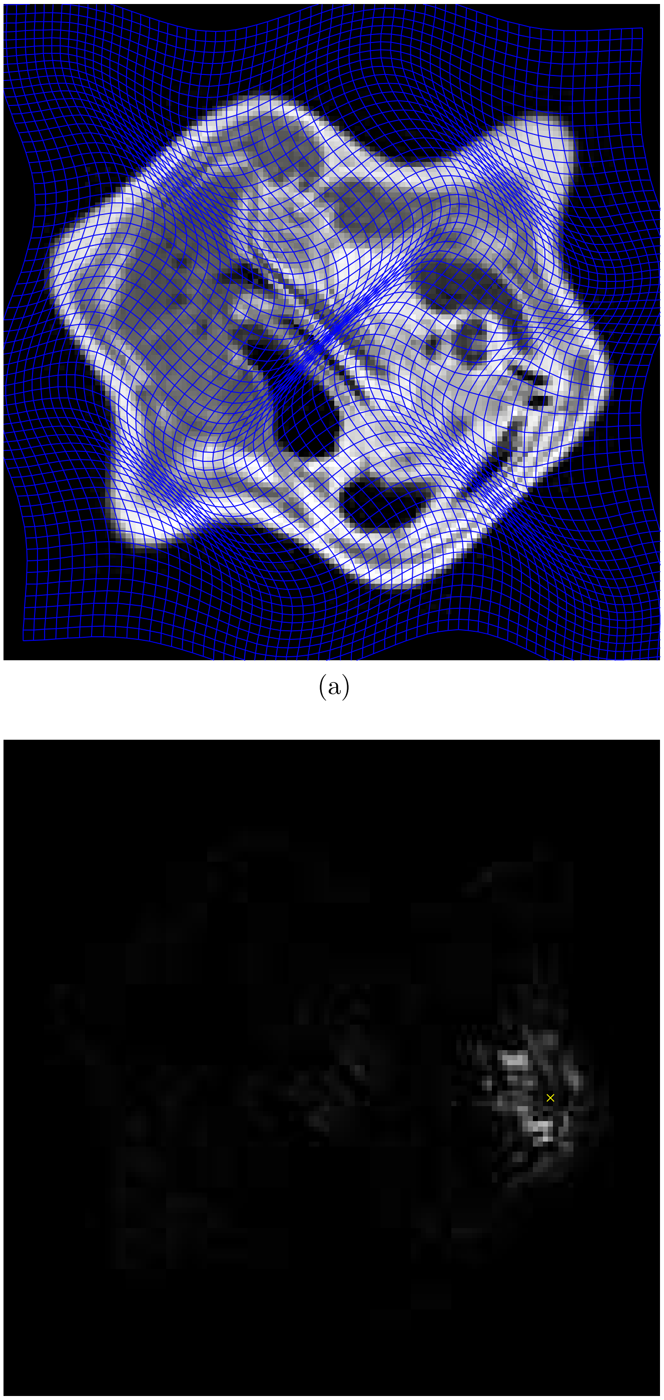
Figure 4.35. Results of experiment 6: Converge rate versus Deformation degree *d*.

4.9.2 Discussion

Compare the fast parametric elastic image registration to our method, the regularity is implied by the use of the div-curl representation of a vector field. This is because both divergence and curl operators involve first order derivatives. Thus, the displacement field is one order smoother than the div and curl fields. Fig. 4.30 through 4.32 indicates our method has better ability to handle larger deformed image. As the previous experiment shows, a stronger regularity can be achieved by using the B-spline coefficients of the divergence and curl fields as the parameters. Fig. (4.33) shows that our formulation seems less prone to grid folding as compared to the Fast parametric elastic image registration method.

4.10 Experiment 7: Proposed Method with User Feedback Input (Interactive Scheme)

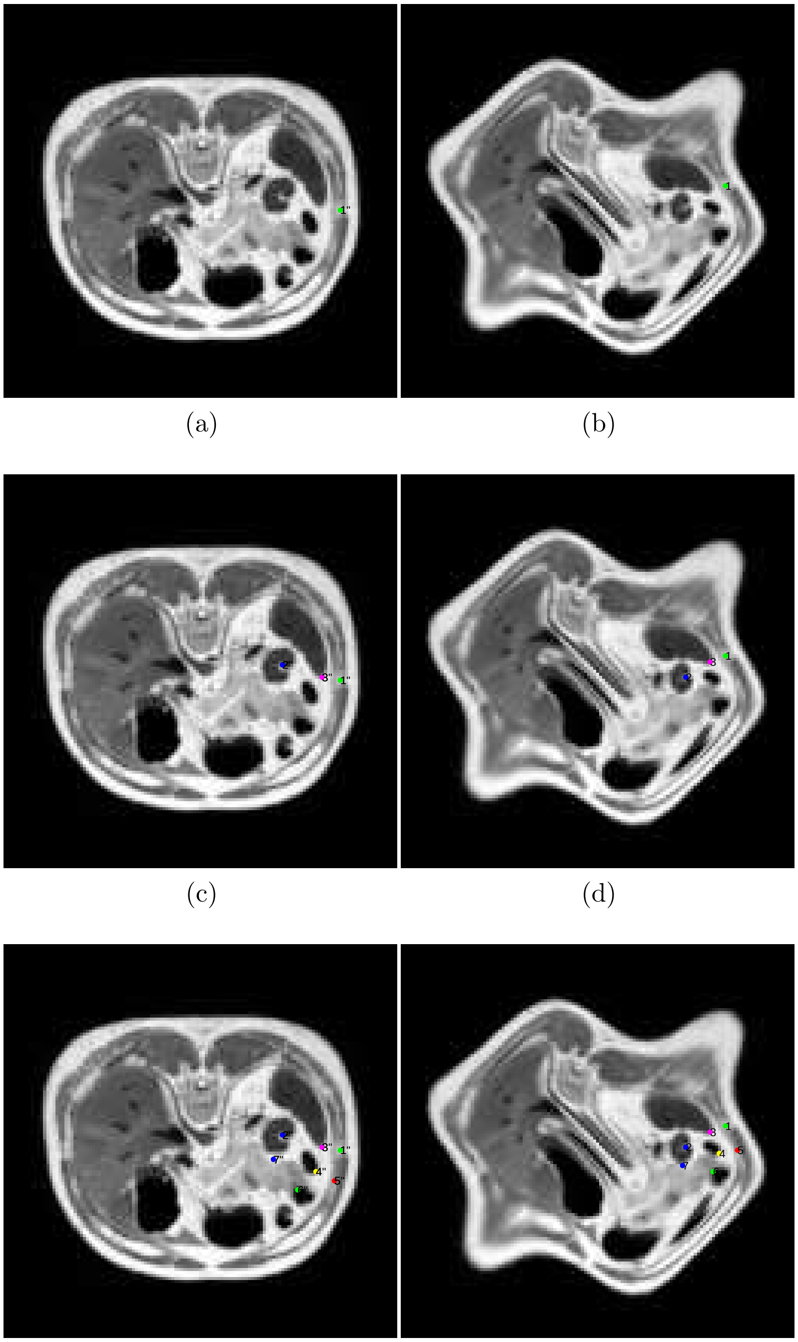
- Purpose: In this experiment, we would like to show if our method cannot register a image set because it falls into a local minimum, the landmark registration is a solution of it.
- Data Set Used: Image sets in 4.1.2 with deformation degree $d = 97$
- Evaluation Criterion: The decision is made accuracy.
- Method Used: Improved scheme with FFT-based div-curl solver and multiresolution embedded.
- Experiment: Fig. 4.36(a) is the grid generated by the proposed method at $d = 97$. Fig. 4.36(b) is the residue image generated by the proposed method at $d = 97$. Based on the residue image, we chose the 1, 3, and 7 landmark points with weights equal to 10. 8, 3 respectively (Shown in Fig. 4.37) to observe the effects of the interactive scheme.



(a)

(b)

Figure 4.36. Results generated by proposed method at degree $d = 97$ (a) Grid (b) Residue Image with x mark indicates the maximum warping index.



(a) (b)

(c) (d)

(e) (f)

Figure 4.37. Landmark points selected on the template image (a) 1 Landmark Point at Template Image (b) 1 Landmark Point at Reference Image (c) 3 Landmark Points at Template Image (d) 3 Landmark Points at Reference Image (e) 7 Landmark Points at Template Image (f) 7 Landmark Points at Reference Image.

4.10.1 Results

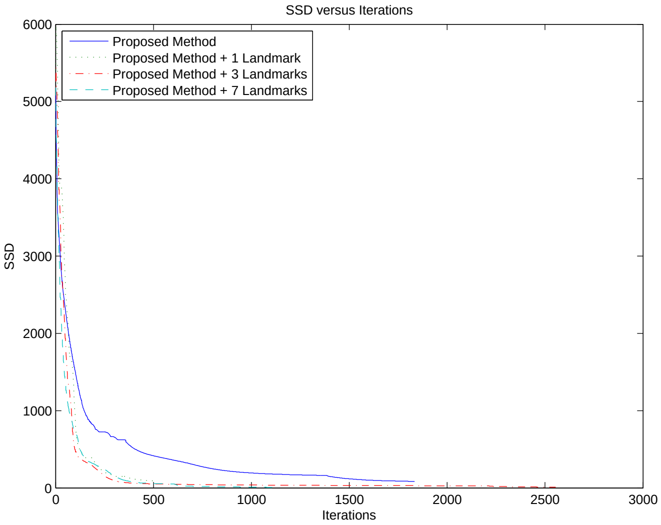


Figure 4.38. Results of experiment 7: SSD versus Iterations.

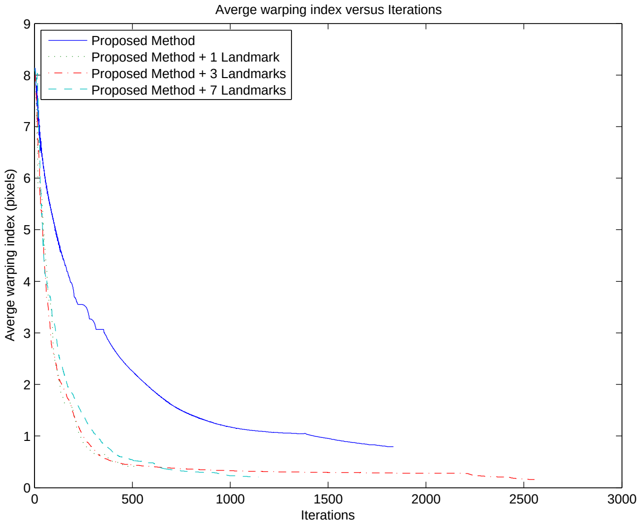


Figure 4.39. Results of experiment 7: Average warping index versus Iterations.

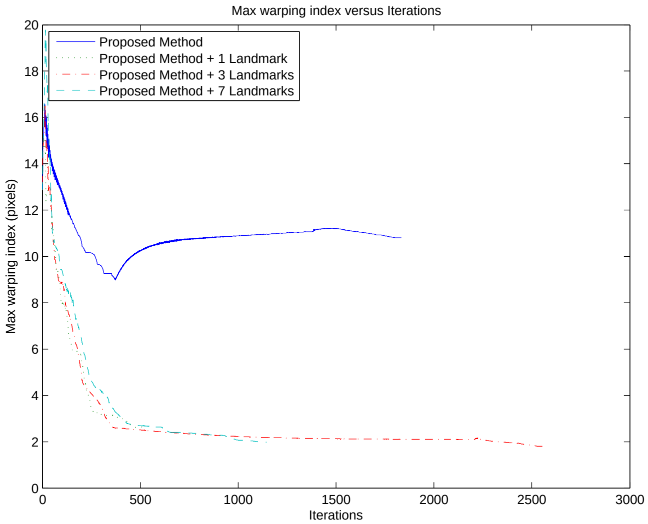


Figure 4.40. Results of experiment 7: Maximum warping index versus Iterations.

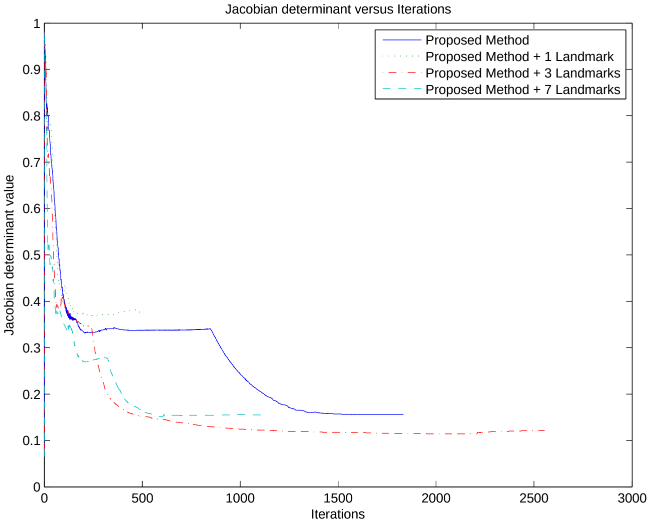
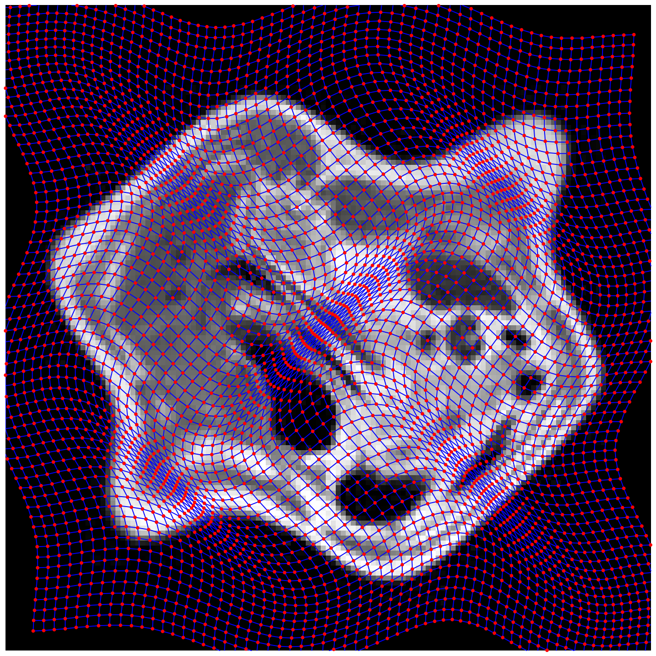


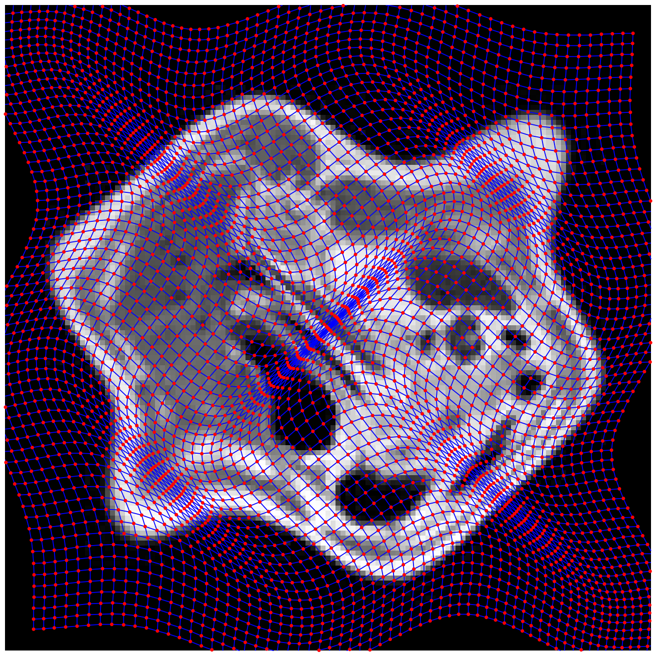
Figure 4.41. Results of experiment 7: Minimum Jacobian value versus Iterations.



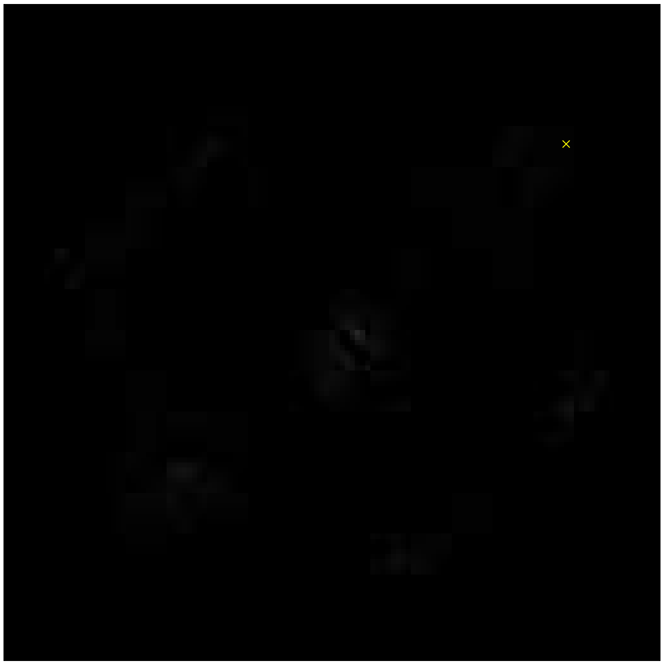
(a)

(b)

Figure 4.42. Results generated by select 1 landmark point (a) Grid (b) Residue Image with x mark indicates the maximum warping index.

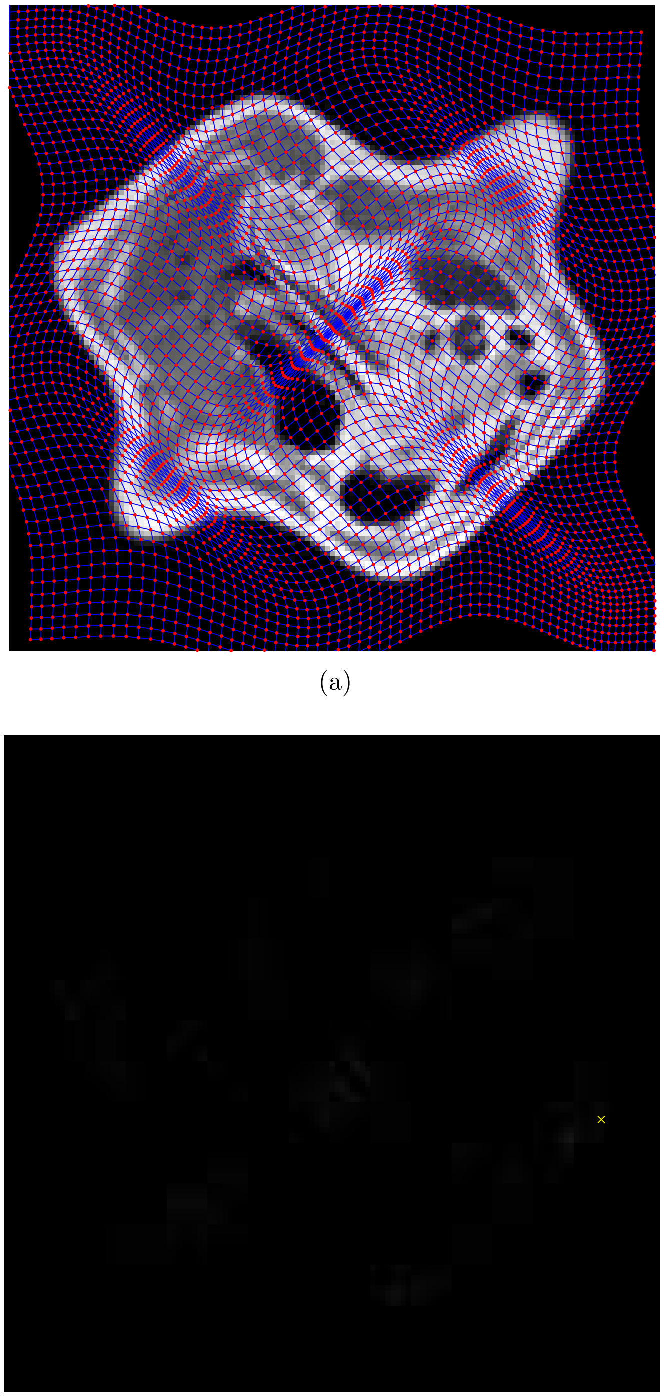


(a)



(b)

Figure 4.43. Results generated by select 3 landmark point (a) Grid (b) Residue Image with x mark indicates the maximum warping index.



(a)

(b)

Figure 4.44. Results generated by select 7 landmark point (a) Grid (b) Residue Image with x mark indicates the maximum warping index.

4.10.2 Discussion

Without landmark points, the proposed method may converge to local minimum. Fig. 4.36 indicates the solution of the proposed method falls into local minimum. With landmark points selected on the residue area, we manually forced the solution avoid those local minimum. Fig. 4.42 through 4.44 shows that, with fewer landmark points, the weights have to be large; On the contrary, more landmark points with smaller weights can also achieve the same goal. The results shows the proposed method with landmark registration has a benefits - it takes the feedback from users to register a image set to his/her desired effects which means it provides an ability to avoid local minimum.

4.11 Experiment 8: An Alternative Optimization

- Purpose: From (3.43), we can see all the divergence and curl parameters are independent, in the other words, these parameters do not have to be adjusted at the same time. In our optimization process, each parameter will be assigned a time step which is to control the effects of gradient information. In this experiment, we would like to show the effective of updating divergence and curl value alternatively.
- Data Set Used: Image sets in 4.1.2
- Evaluation Criterion: The decision is made accuracy, robustness, and efficiency.
- Method Used: Improved scheme with FFT-based div-curl solver and multiresolution embedded.
- Experiment: Currently, the improved scheme is the best method of the proposed methods. It updates the divergence and curl value at the same time. We now re-run the experiment by updating divergence and curl value alternatively.

4.11.1 Results

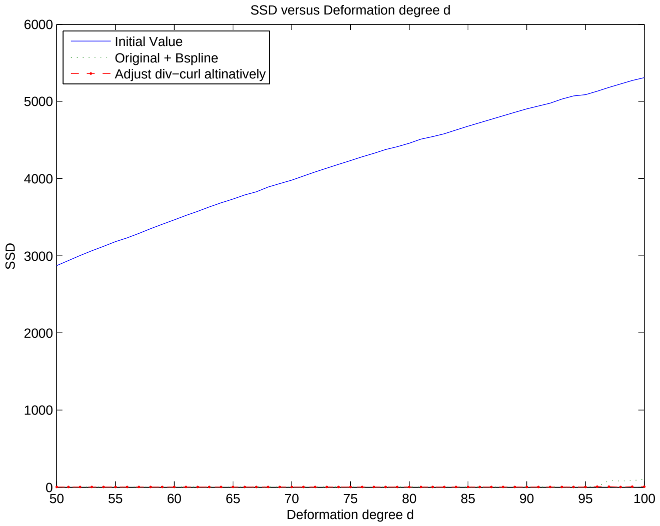


Figure 4.45. Results of experiment 8: SSD versus Deformation degree *d*.

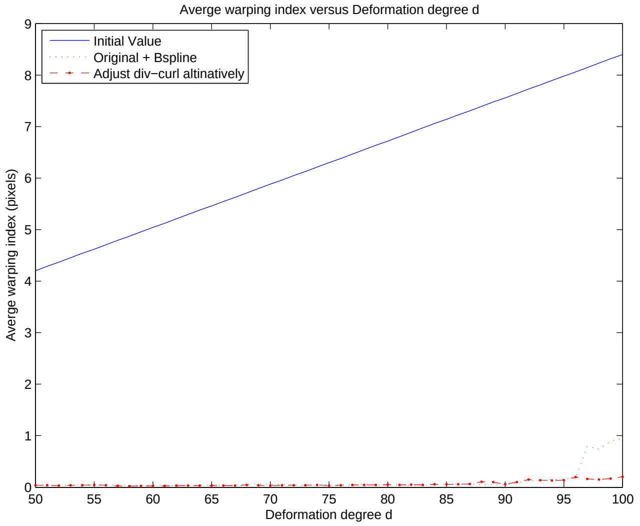


Figure 4.46. Results of experiment 8: Average warping index versus Deformation degree *d*.

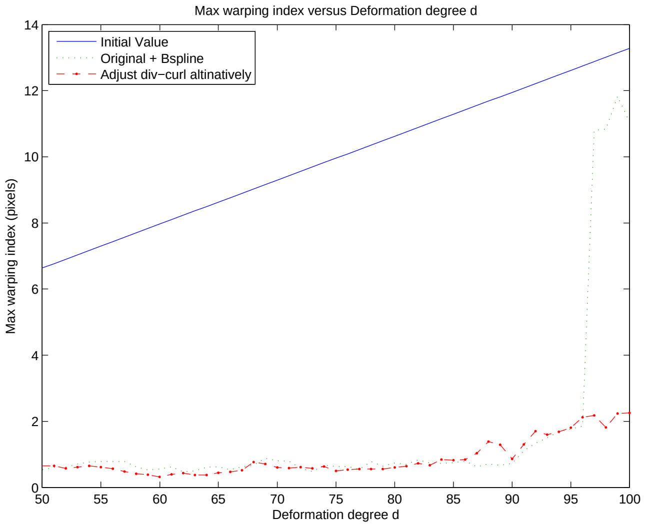


Figure 4.47. Results of experiment 8: Maximum warping index versus Deformation degree d.

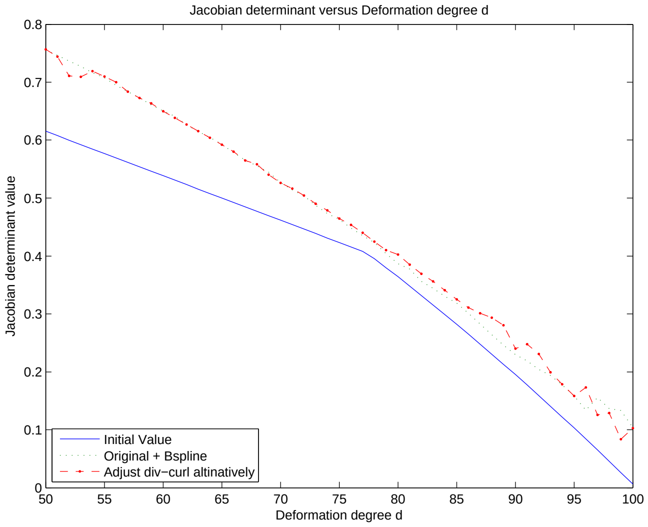


Figure 4.48. Results of experiment 8: Minimum Jacobian value versus Deformation degree d.

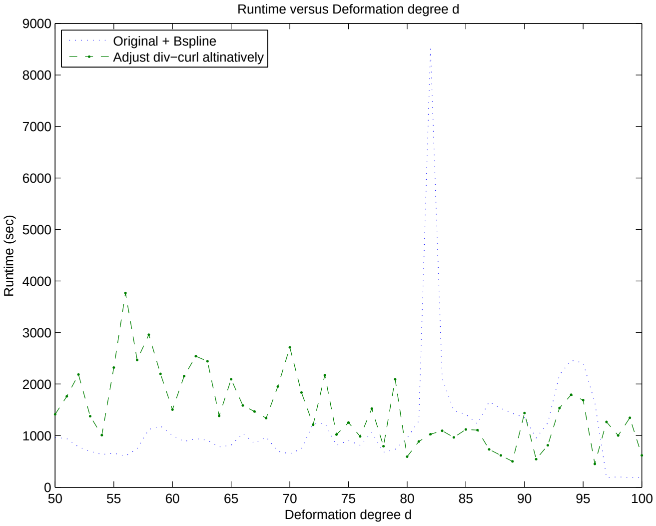


Figure 4.49. Results of experiment 8: Runtime versus Deformation degree *d*.

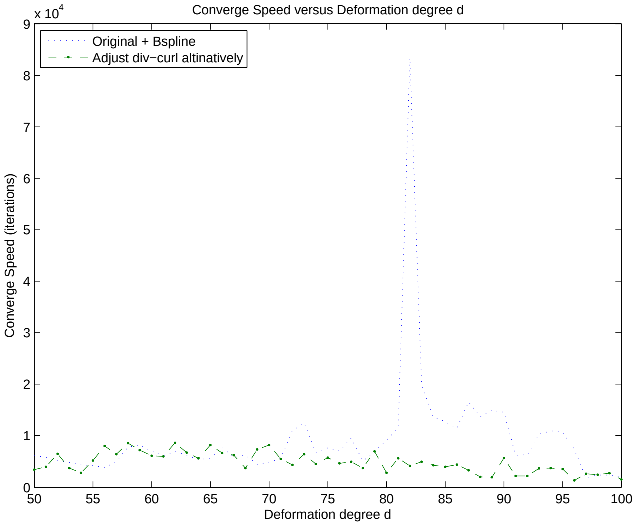


Figure 4.50. Results of experiment 8: Converge rate versus Deformation degree *d*.

4.11.2 Discussion

Compare to the original method, the performance of the new updating method is better in SSD, warping index, and robustness. It is because updating divergence and curl alternatively can avoid the solution falls into a local minimum. Although the complexity of updating divergence and curl alternatively is double from the original method, however, in the real experiment, the runtime is permissible. We think it is because this new updating method drives the solution to the correct place quicker, therefore it does not need more iterations to reach its destination.

4.12 Experiment 9: 3D Clinical Image Test

- Purpose: We have completed all the experiments in 2D. From those experiments, it proves our system has feasible ability to handle the 2D images even with larger deformation. In this experiment, we would like to show the propose system is capable for 3D image volume.
- Data Set Used: Image sets in 4.1.3
- Evaluation Criterion: The decision is made accuracy, robustness, and efficiency.
- Method Used: Improved scheme with FFT-based div-curl solver and multiresolution embedded.
- Experiment: Because of the image size, the multiresolution was run from knot spacing $h = 8$ to $h = 2$.

4.12.1 Results

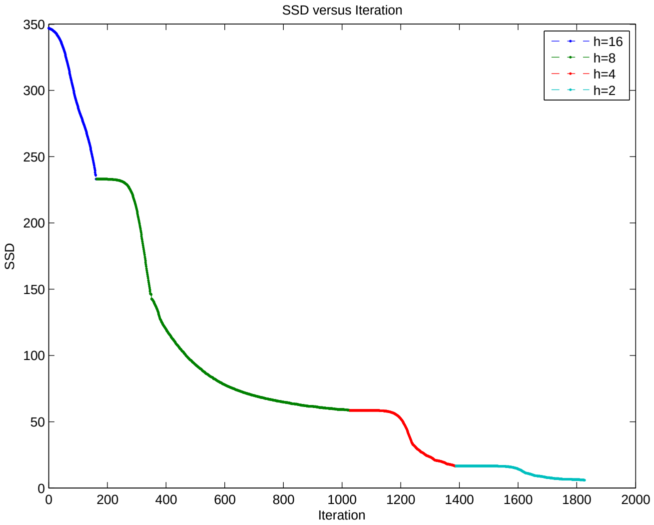


Figure 4.51. Results of experiment 9: SSD versus Deformation degree d.

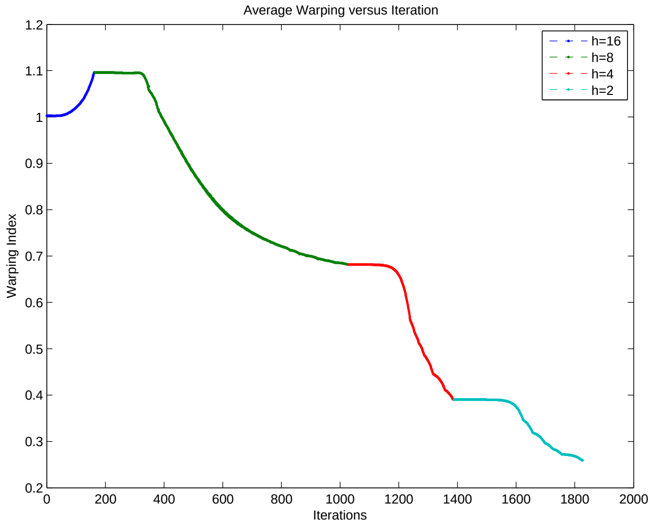


Figure 4.52. Results of experiment 9: Average warping index versus Deformation degree d.

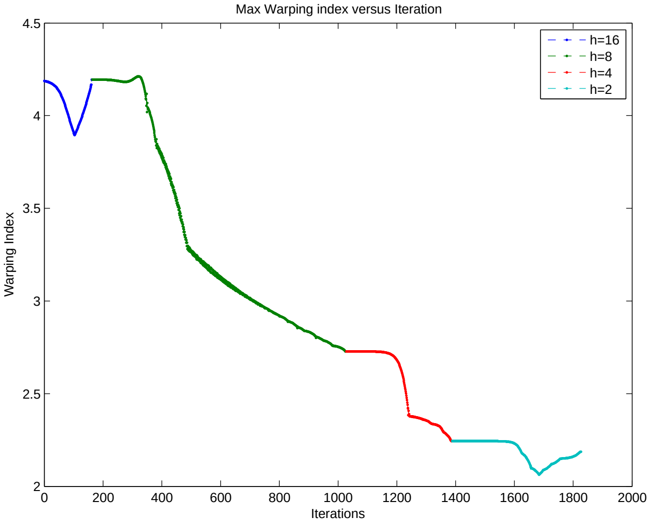


Figure 4.53. Results of experiment 9: Maximum warping index versus Deformation degree *d*.

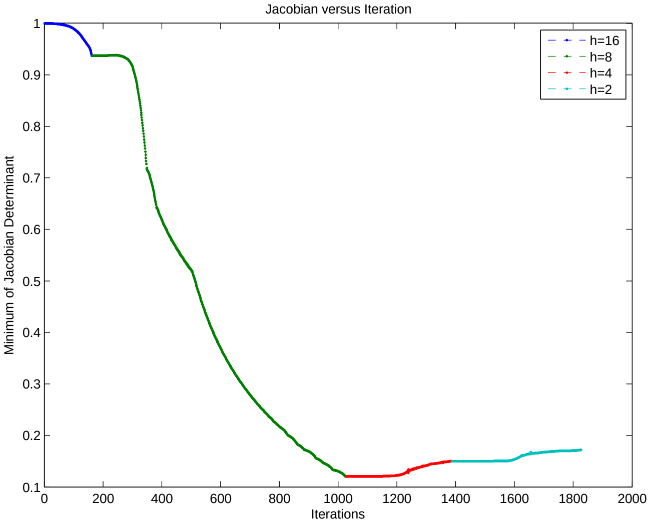
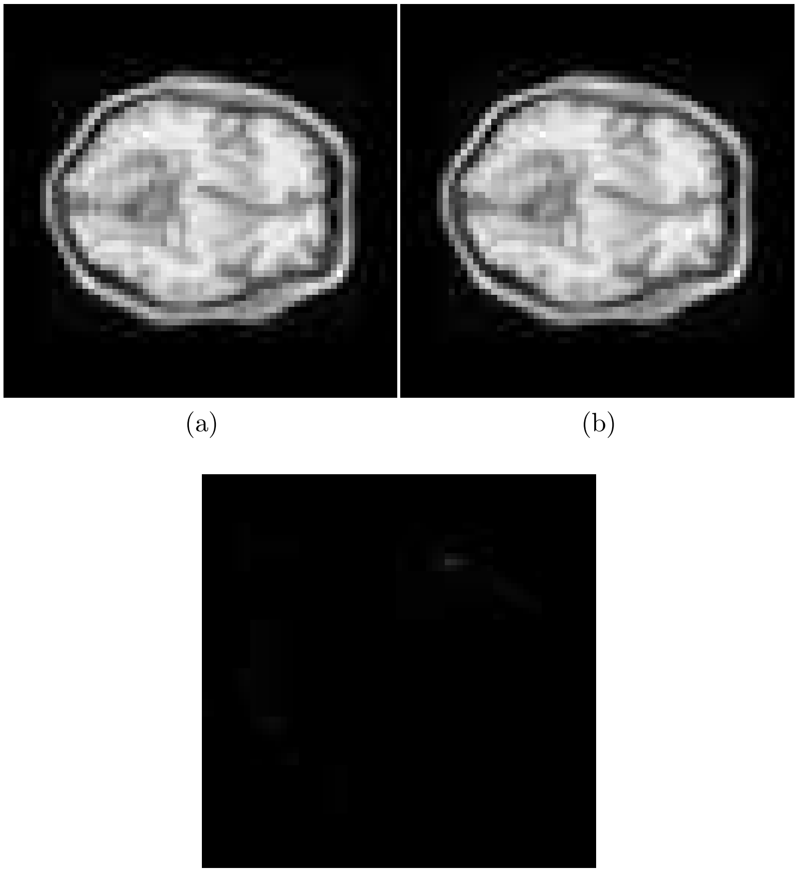


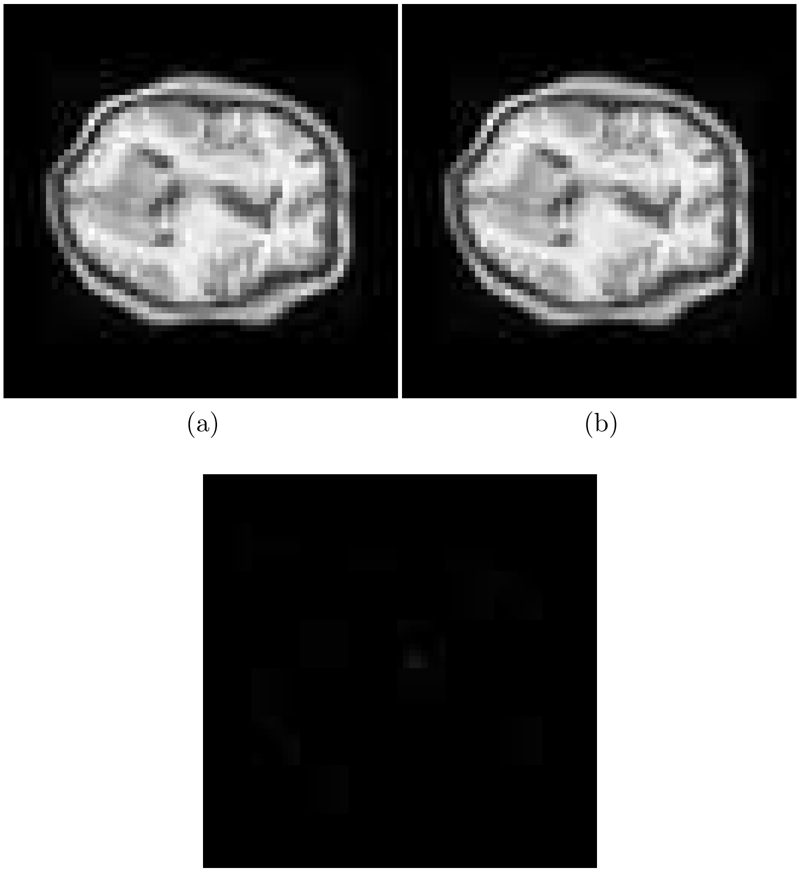
Figure 4.54. Results of experiment 9: Minimum Jacobian value versus Deformation degree *d*.



(a) (b)

(c)

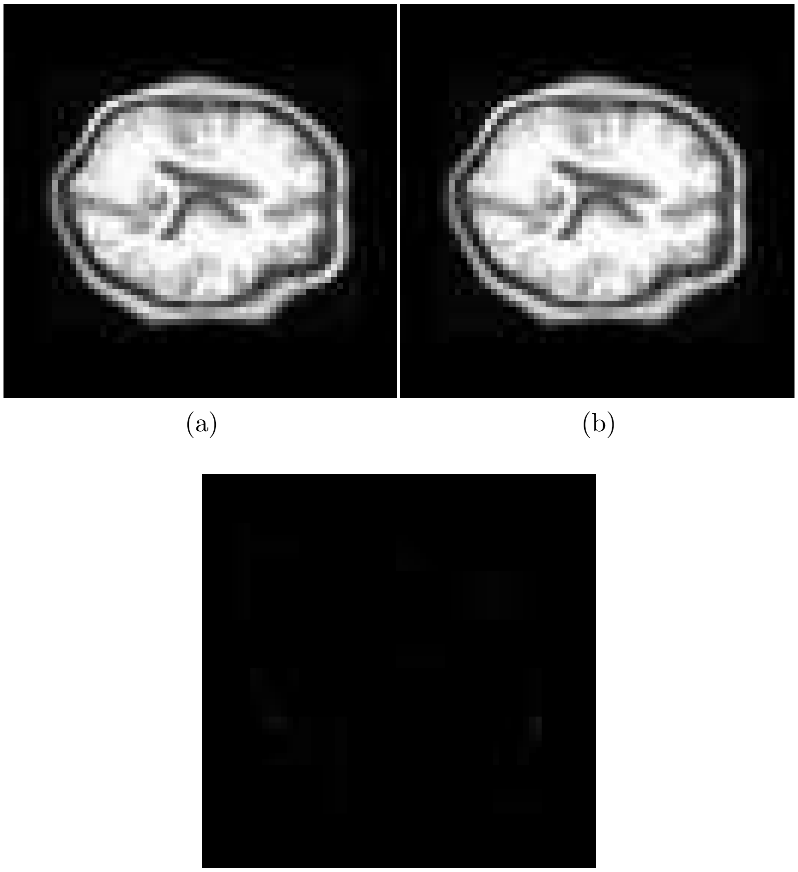
Figure 4.55. The 17^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.



(a) (b)

(c)

Figure 4.56. The 33^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.



(a) (b)

(c)

Figure 4.57. The 45^{rd} slices of 3D brain MRI image vloume (a) Template Image T (b) Reference Image R and (c) Residue Image.

4.12.2 Discussion

Traditionally, rigid registration is applied before a non-rigid image registration process to increase the performance. However, the objective of above 3D experiment is to test the proposed method ultimately, therefore, the image set was warped on three axis based on a sine function. Based on this reason, it is inappropriate to use a rigid image registration before the proposed method. From the results, we observed the proposed method can register a image volume to a complex warped image volume very well. The value SSD is from 349 down to 15 at $h = 4$. The SSD can achieve deeper if we run one more level multiresolution ($h = 2$), however, it is a trade-off between accuracy (fig. 4.53 shows the max warping index jumps a bit at $h = 2$) and runtime. The volume preserved well if we observe Fig. 4.54.

CHAPTER 5

SUMMARY AND FUTURE WORKS

The major contribution of this dissertation is we developed a fully automatic parametric non-rigid image registration algorithm based on Helmholtzs theorem. The parameters are the divergence and curl values of the displacement field at each grid point. The relation to the fast parametric elastic registration method [10] and the parametric non-rigid image registration based on Helmholtz's decomposition method [13] were discussed.

The advantage over [10] is that our method is less prone to grid folding. This is because both divergence and curl operators involve first order derivatives. Thus, the displacement eld is one order smoother than the div and curl fields. Though the other work [13] also used a div-curl system, however, their method leads to a very complicated mathematic model and cannot be generalized to 3-D case easily.

Though it is hard to devise a gradient decent optimization of the proposed method in beginning, we presented a inverse filter based div-curl solver to link the missing puzzle. The inverse filter based div-curl solver can be seen as a support kernel which helps us to figure out the relation between the similarity measure and the parameters - divergence and curl. Although the inverse filter based div-curl solver is an approximation in our case, it also leads us to find a more accurate solver - FFT-based div-curl solver.

Several extensions based on the proposed method were developed. Instead optimizing div-curl parameters, we substituted it to the Poisson parameters. The complexity is less than the proposed method, however, it takes much longer to reach the correct solution. It indirectly proves the Poisson equation that composed by a div-curl system can provides more accurate results. To prevent grid folding, we tried to strengthen the regularity by substituting the linear interpolation to B-spline interpolation. This method not only bring less chance to occur grid folding, but also lead the solution more accurately. The proposed method with B-spline interpolation happened to be our best method which we used to compare with other methods. We also devised a landmark registration method based on the best proposed method, which can register two images based on the users feedbacks.

We have test the proposed methods by various medical images in 2-D and 3-D. The experiment results indicate the proposed methods can handle all of them. In the experiments, we also illustrated the proposed methods can overcome the aperture problem which other methods may not have this ability.

Though the best proposed method has B-spline to strengthen the regularity, our method still cannot guarantee a folding-free grid. In the future, we will try to use the method in [24] to solve this issue even though it is rarely occurred Also, we would like to substitute our similarity to the mutual information (MI) [7, 25], however, the gradient information of MI ($\frac{\partial MI}{\partial f}$) is not easy to be obtained. We believe our method is not limited to the medical image, we will try to apply our method to various applications.

REFERENCES

[1] S. Amador Kane, *Introduction to physics in modern medicine*. London: Taylor Francis, 2003. [Online]. Available: http://www.loc.gov/catdir/enhancements/fy0646/2006276435-d.html

[2] J. B. Campbell, *Introduction to remote sensing*, 4th ed. New York: Guildford Press, 2007. [Online]. Available: http://www.loc.gov/catdir/enhancements/fy0664/2006022914-t.html

[3] J. Modersitzki, *Numerical Methods for Image Registration*, ser. Numerical Mathematics and Scientific Computation. Oxford University Press, January 2004.

[4] P. J. Narayanan, S. K. Nayar, and H.-Y. Shum, *Computer vision: ACCV 2006 : 7th Asian Conference on Computer Vision, Hyderabad, India, January 13-16, 2006 : proceedings*, ser. Lecture notes in computer science. Berlin: Springer, 2006, vol. 3851-3852.

[5] R. R. Beichel and M. Sonka, *Computer vision approaches to medical image analysis: second international ECCV workshop, CVAMIA 2006, Graz, Austria, May 12, 2006 : revised papers*. Berlin: Springer, 2006, vol. 4241. [Online]. Available: http://www.loc.gov/catdir/enhancements/fy0825/2006933642-d.html

[6] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image Vision Comput.*, vol. 21, no. 11, pp. 977–1000, 2003.

[7] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Mutual information based registration of medical images: A survey," *IEEE Trans. Med. Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[8] C. Broit, “Optimal registration of deformed images,” Ph.D. dissertation, University of Pennsylvania, Philadelphia, PA, USA, 1981.

[9] G. E. Christensen, R. D. Rabbitt, and M. I. Miller, “Deformable templates using large deformation kinematics,” *IEEE Transactions on Image Processing*, vol. 5, no. 10, pp. 1435–1447, 1996.

[10] J. Kybic and M. Unser, “Fast parametric elastic image registration,” *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1427–1442, November 2003.

[11] M. UNSER, A. ALDROUBI, and M. EDEN, “B-spline signal-processing .1. theory,” *Ieee Transactions On Signal Processing*, vol. 41, no. 2, pp. 821–833, FEB 1993.

[12] ——, “B-spline signal-processing .2. efficient design and applications,” *Ieee Transactions On Signal Processing*, vol. 41, no. 2, pp. 834–848, FEB 1993.

[13] A. Cuzol, P. Hellier, and É. Mémin, “A novel parametric method for non-rigid image registration,” in *IPMI*, 2005, pp. 456–467.

[14] C. DK, *Field and wave electromagnetics.* Addison-Wesley, 1989.

[15] J.-Z. Wu, H.-Y. Ma, and M.-D. Zhou, *Vorticity and vortex dynamics.* Berlin: Springer, 2006. [Online]. Available: http://www.loc.gov/catdir/enhancements/fy0663/2005938844-d.html

[16] C. Canale, *Numerical Methods for Engineers*, 5th ed. McGraw Hill, 2006.

[17] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition.* Philadelphia, PA: SIAM, 1994.

[18] X. Xu and R. D. Dony, “Fast fluid registration using inverse filtering for non-rigid image registration,” in *ISBI*, 2006, pp. 470–473.

[19] M. Bro-Nielsen and C. Gramkow, "Fast fluid registration of medical images," in *Visualization in Biomedical Computing. 4th International Conference, VBC '96 Proceedings*, 1996, pp. 267–276.

[20] (1996, March) Solving the discrete poisson equation using jacobi, sor, conjugate gradients, and the fft. [Online]. Available: http://www.cs.berkeley.edu/ demmel/cs267/lecture24/lecture24.html

[21] F. L. Bookstein, "Principal warps: thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, 1989.

[22] *Large-Deformation Image Registration Using Fluid Landmarks.* Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium, 2000.

[23] H. mei Chen, "Nonrigid image registration using adaptive grid generation: preliminary results," in *Fourth IEEE International Symposium of Biomedical Imaging (ISBI '07)*, Metro Washington DC, USA, April 2007.

[24] T. Lin, H. Chen, H. Hsiao, and C. Hsieh, "Variational non-rigid image registration using regridding," in *GVGIP 2008.* The 21st IPPR Conference on Computer Vision Graphics and Image Processing, 8/24 - 8/26 2008.

[25] S. Heldmann, O. Mahnke, D. Potts, J. Modersitzki, and B. Fischer, "Fast computation of mutual information in a variational image registration approach," in *Bildverarbeitung für die Medizin*, 2004, pp. 448–452.

BIOGRAPHICAL STATEMENT

Hsi-Yue S. Hsiao received his B.S. and M.S. degree in Computer Science in 2002 and 2005 respectively from The Oklahoma State University at Stillwater, Oklahoma, United State, his Ph.D. degree in Computer Science and Engineering in 2008 from The University of Texas at Arlington, Texas, United States. His research interests include Networking, Artificial Intelligence, and medical image processing. He is a member of the ACM.