REGRIDDING IN NONRIGID IMAGE REGISTRATION

by

TING-HUNG LIN

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

ACKNOWLEDGEMENTS

I would like to thank Dr. Hua-Mei Chen for his invaluable guidance, patience and thorough advice throughout the course of my research and for his passion and dedication to the field of image processing. My gratitude also goes to Dr. Liao, Dr. Gao and Dr. Walker for giving their time to serve on my defense committee. Special thanks belong to my parents who have been so supportive and given me confidence to pursue my dream. Without their love, encouragement and guidance I would not have accomplished as much as I have. Finally, the deepest gratitude to my wife Pei-Wen, who has been with me and taking care of our daughter Olivia in the whole journey.

December, 2008

ABSTRACT

REGRIDDING IN NONRIGID IMAGE REGISTRATION

TING-HUNG LIN, Ph.D.

The University of Texas at Arlington, 2008

Supervising Professor: Hua-Mei Chen, Jean Gao

Regridding was first introduced in viscous fluid registration for preventing folding of the transformation and for maintaining the admissible deformation field in large-deformation nonrigid image registration applications. We investigated the application of regridding to leading nonrigid image registration algorithms, including elastic, fluid, diffusion, curvature, and demons algorithms, and compared the performance and accuracy in each case.

We also introduce a grid repairing mechanism based on the adaptive grid-generation method to prevent the transformation from folding. The grid repairing method can be used in conjunction with the proposed regridding scheme to set bounds on the Jacobian determinant of the transformation. We showed that our regridding and grid repairing method can outperform the original registration algorithms, particularly in large-deformation applications. In this dissertation, we also explain how the proposed method can improve the efficiencyability of the original registration algorithms for large-deformation applications and how the grid repairing method can be embedded in these algorithms.

TABLE OF CONTENTS

Chapter

LIST OF FIGURES

Figure Page

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

Image registration is a very fundamental task in the image processing domain. The goal is to find an optimal geometric transformation between two images, such that the two images become geometrically similar. The demand to register images has arisen in many fields, for example, medical imaging [1–4] and remote sensing [5–7]. In many applications like the respiratory motion, cardiac motion, and deformation-based morphometry [8], nonrigid image registration is required. Most nonrigid image registration algorithms show good results because of the integration of regularization constraints [9, 10]. However the use of regularziation constraints poses another difficulty. Using a strong regularization term may limit the flexibility of registration. On the other hand, a weak regularization term can't prevent the transformation from becoming singular.

Regridding was first introduced in [11] for preventing folding of the transformation and for maintaining the admissible deformation field in large-deformation nonrigid image registration applications. It is performed whenever the minimum Jacobin determinant of the deformation field is below a pre-defined threshold (0.5 in [11]). According to [11], regridding, however, will increase the numerical precision errors of the transformation due to the successive transformation compositions. However, such worry is not necessary as far as registration accuracy is concerned. Also in [11], the transformation concatenation mechanism was wrongly presented and consequentially it hindered the idea of regridding through transformation concatenation. In this dissertation, we correct the transformation concatenation mechanism in [11] and use linear interpolation to realize transformation concatenation to ensure that each propagated template is always deformed from the original template. The transformation concatenation can't guarantee that the topology of the underlying grid is preserved. To overcome this problem, we devised a grid repairing mechanism and integrate it into the proposed regridding method to preserve the image topology. This method is based on the deformation based grid generation proposed in [12–14], which is described in Chapter 2. With the grid repairing method, we are able to control the Jacobian determinant of the deformation field. Image topology is automatically preserved if the Jacobian determinant field is maintained to be strictly positive. In addition, we show that it is a very powerful tool to improve the performance of many existing nonrigid image registration algorithms in large-deformation applications. The topology preserving by grid repairing method can be applied in many medical imaging applications [8, 15–20]. While J. Ashburner et al. used the deformation filed to identify the anatomical differences [8], C. Gaser et al. used the change of volume of deformation field for schizophrenia research [17]. Hence, the deformation filed generated by nonrigid image registration method is quite important and should be unfolded/non-tangled.

In this dissertation, we adopted five different nonrigid image registration algorithms: elastic, fluid, diffusion, curvature and demons algorithms and show how to incorporate regrididng with/without grid repairing into these registration algorithms and compare the results. This dissertation is organized as follows. In Chapter 2, we review some major nonrigid image registration methods: elastic registration, fluid registration, diffusion registration, curvature registration, and demons registration. Implementation of regridding in [11], transformation concatenation, and adaptive grid generation method are also reviewed in this section. In Chapter 3, we introduce the proposed regridding method and grid repairing by using adaptive grid generation method. The implementation of how to integrate the regridding and grid repairing method with the registration methods adopted in this dissertation is also described in this section. Experimental results using synthetic data and simulated medical images are presented in Chapter 4. Finally, discussion and conclusion are given in Section 5 together with directions for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Variational Based Image Registration

Given two images, a reference image R and a template image T, the task is to find a transformation (global / local) from R onto T in such a way that the transformed template matches the reference image. The transformation can be described by a displacement field $u : \Omega \to \Omega$ such that $T_u \equiv T(x + u(x))$ is similar to R in the geometrical sense. To find such a mapping $u = (u_1, ..., u_d)$, d is the dimension of the images, in variational framework, a cost functional consisting of two terms is devised,

$$J[u] = D[R, T; u] + \alpha S[u] \tag{2.1}$$

where D represents a distance measure whose gradient is used as the external force and S is called the smoother, which determins the smoothness of the displacement field u. The gradient of S is considered as the internal force. $\alpha > 0$ is a parameter to control the strength of the smoother term. Registration is achieved when the external force is balanced by the internal force. Among various choices for the functional D in Eq. (2.1), sum of squared differences (SSD) is widely used for monomodal applications and is used in this dissertation. It is defined as,

$$D[R, T; u] = \frac{1}{2} \int_\Omega (T(x - u(x)) - R(x))^2 dx \tag{2.2}$$

Different regularization terms in Eq. (2.1) lead to different variational image registration algorithms.

2.1.1 Elastic Registration

Elastic registration is first introduced by Broit [21]. By remodeling the elastic registration to fit the variational approach, the smoother shown in Eq. (2.3) is introduced in [10, 22]. Where μ and λ are the so-called *Lamé* constants and the resulting registration method is exactly the same as the original elastic registration [21]. The smoother is a *linearized elastic potential* of the displacement u. This smoother impose penalties based on the first order derivative of the deformation field.

$$S^{elastic}[u] = \int_{\Omega} \frac{\mu}{4} \sum_{j,k=1}^{d} (\partial_{x_j} u_k + \partial_{x_k} u_j)^2 + \frac{\lambda}{2} (\text{div } u)^2 dx \tag{2.3}$$

It was suggested in [22] to use a regularized incremental update to the displacement u. And the final deformation is given by $x + \sum_{i=1}^{k} u^{(i)(x)}$, where k denotes the number of iteration. We named this modification as a modified version of elastic registration.

2.1.2 Fluid Registration

Christensen [11] proposed to use instead a viscous fluid model for the deformation. His derivation was based on a specific linearization of the *Navier Stokes* equation. Actually, as it turns out, one may obtain Christensen's approach by invoking the elastic potential of the velocity v of the displacement field

$$S^{fluid}[u] := S^{elastic}[v] \tag{2.4}$$

which might be viewed as a visco-elastic model [10, 23]. By introducing an artificial time t, the velocity and the transformation are related via the material derivative

$$v(x,t) = u(x,t) + \nabla u(x,t) v(x,t) \tag{2.5}$$

The combination of D and $S^{fluid}[u]$ is called fluid matching or fluid registration.

2.1.3 Diffusion Registration

The smoother of diffusion registration is introduced in [9] where

$$S^{diffusion}[u] = \frac{1}{2}\sum_{l=1}^{d}\int_{\Omega} \parallel \nabla u_l \parallel^2 dx \tag{2.6}$$

Its implementation is based on a finite difference approximation of a diffusion like equation. The main advantage of this registration is the speed. It can be implemented by AOS method [9].

2.1.4 Curvature Registration

The smoother of curvature registration is introduced in [10], and it is based on the following smoothing term,

$$S^{curvature}[u] = \frac{1}{2}\sum_{l=1}^{d}\int_{\Omega}(\Delta u_l)^2 dx \tag{2.7}$$

where Δ is the Laplace operator. In [10], it is empathized that this scheme does not require an additional affine linear pre-registration step to be successful.

2.2 Demons Registration

The Demons algorithm, originally proposed by Thirion [15], is a deformable registration algorithm that is widely used to match medical images. Demons is based on the optical flow method [24] which is used to find small deformations in temporal sequences of images. The optical flow method finds a displacement field that deforms the target image, T, so that it is matched with the reference image, R. The basic hypothesis of optical flow is that intensities are constant between T and R, which leads to the following optical flow equation for a given position x:

$$\overrightarrow{v} \cdot \overrightarrow{\nabla} R = T(x) - R(x) \tag{2.8}$$

Here, $\overrightarrow{\nabla} R$ denotes the intensity gradient vector of R. In addition, $T(x)$ and $R(x)$ are the intensity values of T and R at x respectively. Thirion follows an iterative approach and proposes the Demons algorithm based on optical flow [15]. This iterative algorithm alternates between computation of additional displacement field and regularization of the total displacement field until convergence. The additional displacement obtained by demons is calculated as following,

$$\overrightarrow{v} = \frac{(T(x) - R(x))\overrightarrow{\nabla} R(x)}{(\overrightarrow{\nabla} R(x))^2 + (T(x) - R(x))^2} \tag{2.9}$$

In optical flow, $\overrightarrow{v}$ is considered to be a velocity because the images are two successive time frames: That is, $\overrightarrow{v}$ is the displacement during the time interval between the two image frames. In fact, when comparing images of two different sources, there is no such temporal consideration and it is more general to consider $\overrightarrow{v}$ as being simply a displacement. The problem is that v is in general not smooth. To project the update onto a smooth space by convolving with a Gauss filter is suggested in [15] and this is the regularization mechanism adopted in demons algorithm.

2.3 Regridding

To deal with large-deformation application and prevent the grid from being singular, Christensen introduced the concept of regridding in his fluid registration model [11]. This method computes the determinant of Jacobian of the deformation at each step. As long as the determinant of Jacobian is larger than a certain threshold, there is no invertibility problem and the additive scheme is used. When the Jacobian becomes anywhere smaller than a certain threshold, to avoid grid folding, regridding of the deformed template image is applied to generate a new template, setting the incremental displacement field to zero. The total deformation is the concatenation of the deformation fields associated with each propagated template.

Regridding can be defined as a process of reinitializing the deformed grid at an intermediate stage of an image registration process. Assume regridding is taken place at an intermediate stage when $\varphi = \varphi'$ where $\varphi(x) = x + u(x)$ is the deformation field. At this stage, the two images are $R(x)$ and $T(\varphi' \circ \varphi_0(x))$ where $\varphi_0 = I$ is the identity transformation. After regridding, φ' is stationary and φ_0 continues to vary from I to φ''. Assume regridding is performed again when $\varphi_0 = \varphi''$, we have $R(x)$ and $T(\varphi' \circ \varphi'' \circ \varphi_0(x))$, where φ_0 is again I after regridding and registration continues by varying φ_0. In [11], two algorithms were provided to realize the concept of regridding. In the first algorithm, after the first regridding process, the deformed template is obtained as $T'(x) = T(\varphi'(x))$. When registration continues, the two images to be registered become $R(x)$ and $T'(x)$. After the second regridding process, the deformed template is obtained from $T'(x)$ as $T''(x) = T'(\varphi''(x))$. In this manner, a new template $T''(x)$ is propagated from the previous template $T'(x)$. This method is intuitive and easy for implementation but the interpolation error is also propagated and the image detail is lost after regridding several times. In [11], the second algorithms is able to solve this problem. However, a fatal mistake was found in the original paper and perhap it is the reason why, to our best knowledge, it has never been adopted to realize regridding. The idea of the second algorithm was to obtain the total deformation field by concatenating all incremental deformation fields through interpolation. In this way, the deformed template can always be obtained from the original template. This idea was wrongly expressed in [11] as Eq.(2.10) where $\vec{x}$ is the regular grid, $\vec{U}$ is the incremental deformation field and $\vec{u}$ is the total deformation field, while the correct formula should be Eq. (2.11). That is, it is the total deformation field is obtained by interpolating the incremental deformation field as shown in Eq. (2.10) [11].

$$\vec{u}(\vec{x}, t_i) \triangleq \begin{cases} \vec{U}^{(0)}(\vec{x}, t_i) & 0 \leq i \leq p_1 \\ \\ \vec{U}^{(j)}(\vec{x}, t_i) + \vec{U}^{(j-1)}(\vec{x} - \vec{U}^{(j)}(\vec{x}, t_i), t_{p_j}) & p_j < i \leq p_{j+1} \end{cases} \tag{2.10}$$

In order to correctly concatenate successive deformation fields, the total deformation field obtained in the previous regridding stage that should be interpolated, rather than the incremental deformation field.

$$\vec{u}(\vec{x}, t_i) \triangleq \begin{cases} \vec{U}^{(0)}(\vec{x}, t_i) & 0 \leq i \leq p_1 \\ \\ \vec{U}^{(j)}(\vec{x}, t_i) + \vec{u}(\vec{x} - \vec{U}^{(j)}(\vec{x}, t_i), t_{p_j}) & p_j < i \leq p_{j+1} \end{cases} \tag{2.11}$$

2.4 Deformation Based Grid Generaion

The deformation based grid generation proposed in [12] is to deal with the following problem. Let $\Omega \subset \mathbb{R}^n$ be a bounded open domain. Let $f \in C^1$, $f > 0$, and $\int_\Omega f = |\Omega|$, where $|\Omega|$ is the volume of the solution domain Ω Find a mapping function ϕ from Ω onto itself, such that

$$\begin{aligned} & J\left(\phi\right) \equiv \det \nabla \phi\left(x\right) = f\left(x\right), x \in \Omega \\ & \phi(x) = x \text{ on } \partial\Omega \end{aligned} \tag{2.12}$$

The method proposed in [12] to find such a mapping ϕ is summarized below.

Step 1: Find a vector field $\eta(x)$ that satisfies:

$$\nabla \cdot \eta(x) = f(x) - 1 \tag{2.13}$$

and

$$\eta(x) = 0,\, x \in \partial\Omega \tag{2.14}$$

Step 2: Form a time-varying velocity vector field,

$$V(x,t) = \frac{\eta(x)}{t + (1-t)f(x)},\ t \in [0,1] \tag{2.15}$$

Step 3: Find $\phi_t\left(x\right)$ by solving the following ordinary differential equation (ODE)

$$\frac{d\phi_t(\xi)}{dt} = V_t(\phi_t(\xi)), t \in [0,1] \tag{2.16}$$

then, $\phi\left(\xi\right) = \phi_{t=1}\left(\xi\right)$. The proof of the above approach can be found in [12]. Note that Eq. (2.13) is the result of linearization of Eq. (2.12) around the identity map; *i.e.*, if we let η be a small displacement from the identity mapping Id, then $J(Id+\eta) = f$ implies that η satisfies Eq. (2.13). Eqs. (2.13)-(2.16) map a regular grid into a general grid in the same domain with a transformation whose Jacobian determinant $(\det|J(\phi)|)$ is equal to the specified scalar monitor function f. Theoretically, restricting $f > 0$, no grid folding will result because of the Eq. (2.12) where $J = f > 0$. Notice that the solution ϕ is not unique. This is because in **Step 1**, only the divergence of the vector field η is specified. To obtain an unique η, both of its divergence as well as its curl need to be specified. In this dissertation, **Step 1** is modified to ensure the uniqueness of the solution.

Modified Step 1: Find a vector field $\eta(x)$ that satisfies the following div-curl system:

$$\begin{cases} \nabla \cdot \eta(x) = f(x) - 1 \\ \nabla \times \eta(x) = g(x) \end{cases} \tag{2.17}$$

with null boundary condition $\eta(x) = 0$, $x \in \partial\Omega$, where $g(x)$ is a scalar function in 2D case and a 3D vector function in 3D case specifying the curl of the vector field η.

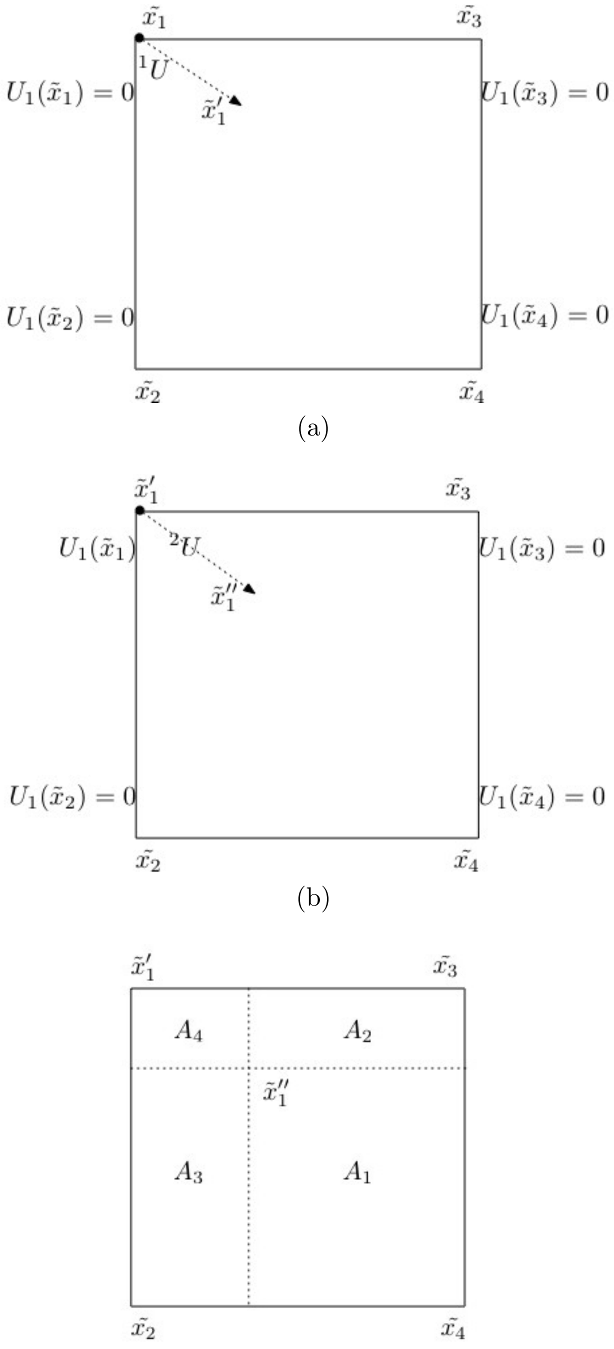
CHAPTER 3

PROPOSED REGRIDDING METHOD

The Algorithm2 in Christensen's work requires an interpolation scheme to realize the transformation concatenation operation when regridding is invoked [11]. Furthermore, no explicit interpolation scheme was suggested in [11] while the quality of the interpolation scheme will affect the quality of the concatenated transformation. More specifically, the topology of the deformable template may not be well preserved. In this chapter, we present a transformation concatenation algorithm based on linear interpolation in this chapter. However, it turns out that it does not preserve the topology of the template although the complete displacement field can be obtained by concatenating all successive displacement fields in this manner. This is due to the limit of linear interpolation scheme. To overcome this instead of devising a "perfect" interpolation scheme for this purpose, which is even more challenging, we then propose a grid repairing mechanism to overcome this problem which is also presented in this chapter.

3.1 Transformation Concatenation

Fig. 3.1a shows a cell of a regular 2D grid composed of four grid points $\tilde{x}_1$, $\tilde{x}_2$, $\tilde{x}_3$, and $\tilde{x}_4$. Each grid point is associated with a displacement vector initially being zero. This is regarded as the initial state before registration. Assume that regridding is activated after an intermediate stage of the registration process when $\tilde{x}_1$ moves to $\tilde{x}_1'$. Denote the displacement of $\tilde{x}_1$ at this stage as ${}^1U$. After regridding, $\tilde{x}_1'$ is restored to the original position $\tilde{x}_1$ and the transformation is set to be identify.



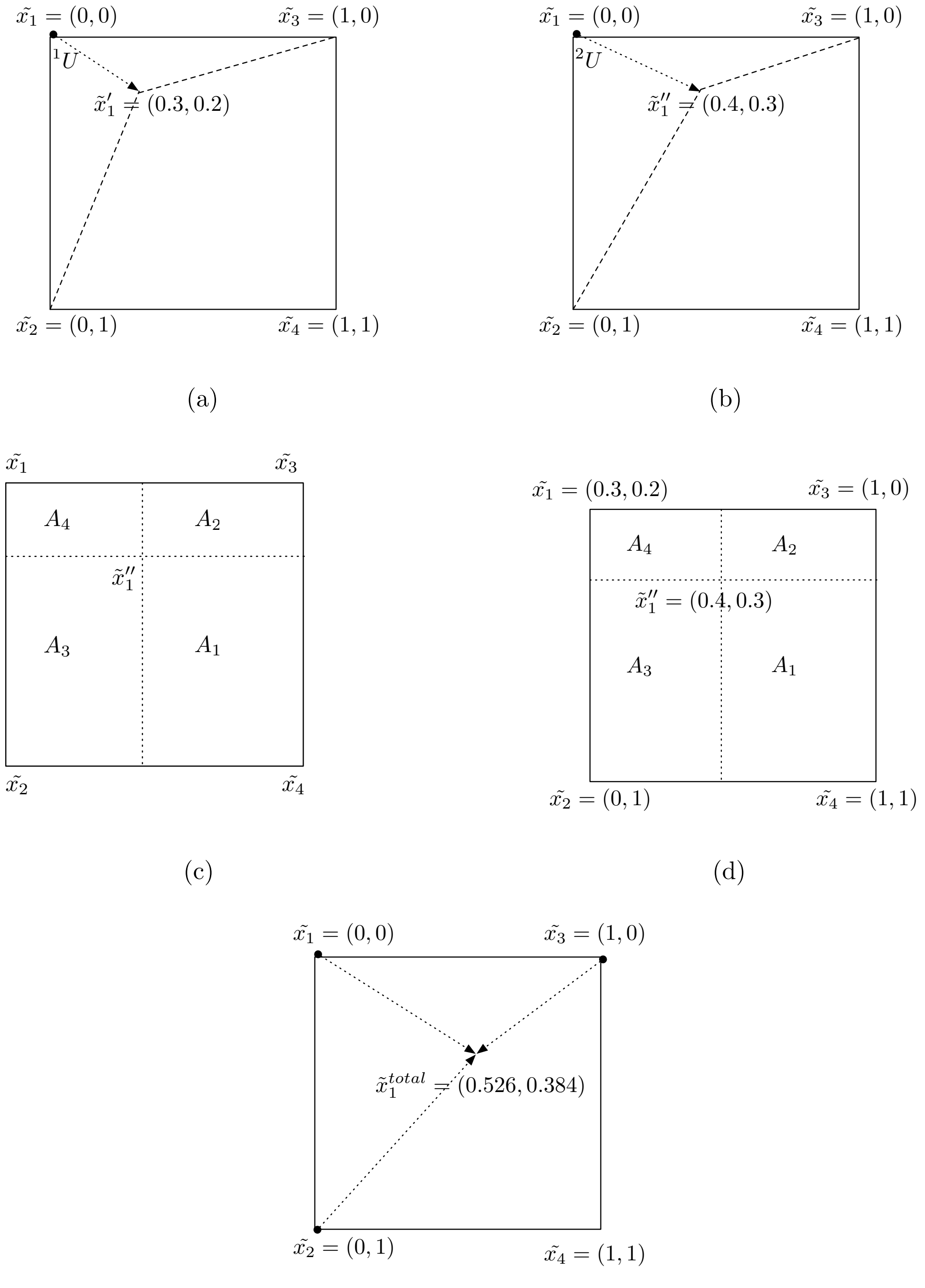
(a)

(b)

(c)

Figure 3.1. (a)A cell of regular 2D grid and corresponding displacements (b)A cell of regular 2D grid and corresponding displacements after regridding (c)The weights of displacement $\tilde{x}_1$.

Let another registration process takes place and brings the point $\tilde{x}_1'$ to new positions $\tilde{x}_1''$, and denote the displacement of $\tilde{x}_1'$ due to this stage of registration process by ${}^2U$. Determining the position of $\tilde{x}_1''$ in the reference frame of the original template is equivalent to composite the displacement ${}^2U$ with ${}^1U$. This can be accomplished by interpolating the displacement ${}^2U$ in Fig. 3.1b to Fig. 3.1a with d-linear interpolation. For example, the displacement of $\tilde{x}_1''$ in the original reference frame is equal to ${}^2U(\tilde{x}_1)+$ $\sum_{i=1}^{4} A_i \cdot {}^1U(\tilde{x}_i)$ as illustrated in Fig. 3.1c by using linear interpolation, where A_1, A_2, A_3, and A_4 are the areas of the sub-cells as indicated.



(a)

(b)

(c)

(d)

(e)

Figure 3.2. Implementation of the alternate regridding approach using linear interpolation (a)A cell of regular 2D grid (b)A cell of regular 2D grid after regridding (c) The weights of displacement $\tilde{x}_1$ (d)The equivalent position of (a) and (b) (e)The final position .

This method is illustrated in the example shown in Fig. 3.2. Fig. 3.2. Fig. 3.2a shows a cell of a regular 2D grid composed of four grid points $\tilde{x}_1$, $\tilde{x}_2$, $\tilde{x}_3$, and $\tilde{x}_4$. Each grid point is associated with a displacement vector initially being zero. Regridding is activated after an intermediate stage of the registration process when $\tilde{x}_1 = (0,0)$ moves to $\tilde{x}_1' = (0.3, 0.2)$. The displacement of $\tilde{x}_1$ is ${}^1U = (0.3, 0.2)$. After regridding, $\tilde{x}_1'$ is restored to the original positions $\tilde{x}_1$ as shown in Fig. 3.2b. Let another registration process takes place and brings the point $\tilde{x}_1'$, which has been restored to the original position $(0,0)$, to a new position $\tilde{x}_1'' = (0.4, 0.3)$, and the displacement of the point $\tilde{x}_1$ is ${}^2U = (0.4, 0.3)$. The position of $\tilde{x}_1''$ in the reference frame also denoted as the $\tilde{x}_1^{total}$ is interpolated by $(0.4, 0.3) + A_1 \cdot (0.3, 0.2) + A_2 \cdot (0,0) + A_3 \cdot (0,0) + A_4 \cdot (0,0)$, where $A_1 = 0.42, A_2 = 0.28, A_3 = 0.18, A_4 = 0.12$. And $\tilde{x}_1^{total} = (0.4, 0.3) + (0.126, 0.084) = (0.526, 0.384)$ An alternative view of this approach is the following. Assume we associate each grid point with its coordinate in the original reference frame, then the coordinates with respect to the original reference frame of any point in the grid ca be obtained through interpolation. This alternative view is illustrated using the same example in Fig. 3.2d.

3.2 Folding Problem by Transformation Concatenation

The regridding method through linear interpolation is able to obtain the complete displacement field by concatenating all successive displacement fields. However, it does not preserve the topology of the deformation in the sense that the Jacobian determinant of the composite transformation may becomes negative even when the Jacobian determinant of each individual mapping is strictly positive. Fig. 3.3 shows an example of the folding problem after the concatenation of two transformations whose Jacobian determinants are both positive. Fig. 3.3a is the original grid. Fig. 3.3b is the grid from Fig. 3.3a after regridding. Assume that point B moves to B' after a

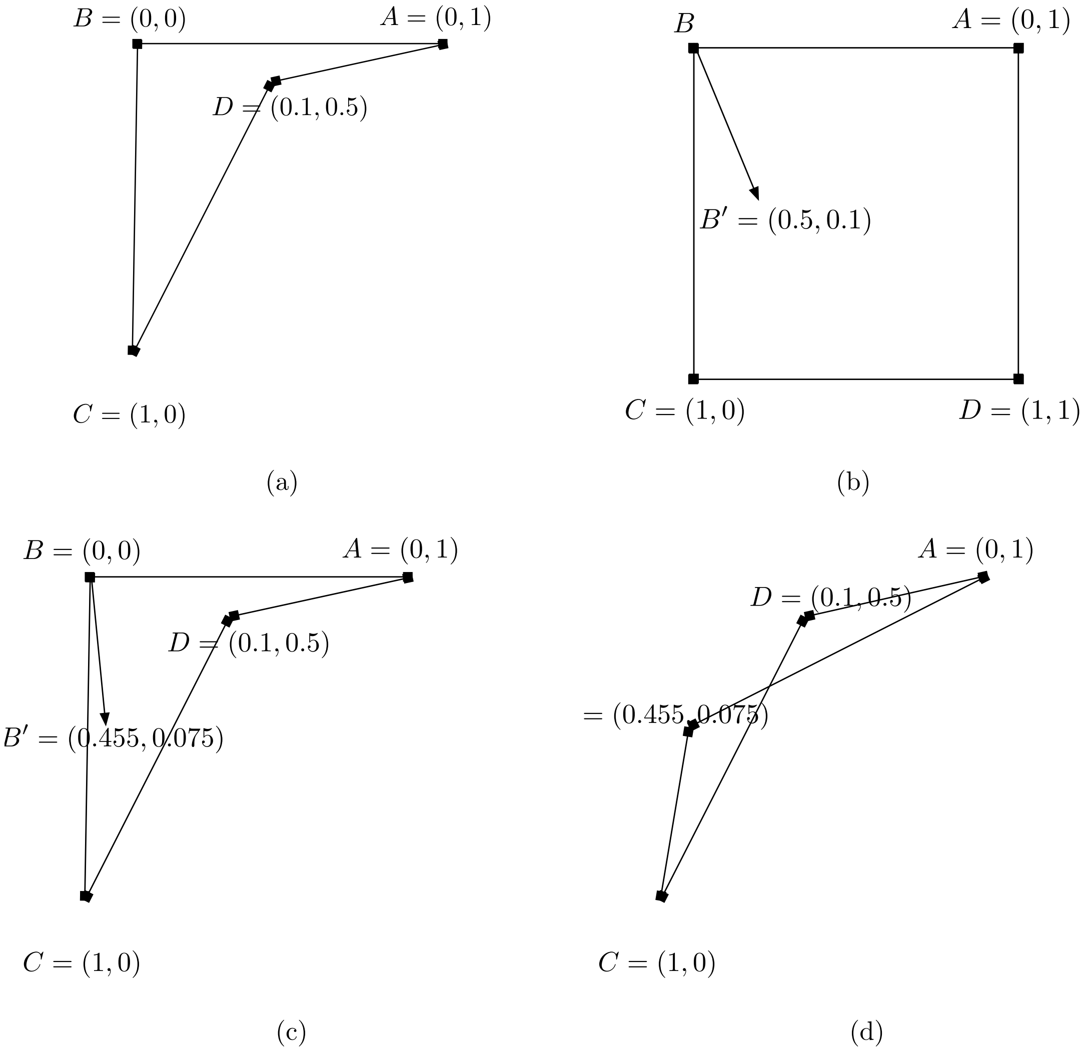


Figure 3.3. An example illustrating the grid folding problem using the proposed regridding method (a)the original grid (b)the grid after regridding (c)the equivalent displacement in the original grid (d)the connection of each points.

deformation. Fig. 3.3c shows the equivalent position of B' in Fig. 3.3a by using the deformation composition method illustrated in Sec. 3.1. The connection of $AB'CD$ results in a folded grid as shown in Fig. 3.3d. To overcome this problem, we developed a grid repairing method that is able to repair an incremental displacement field such that the topology of the template is preserved after being transformed by the composite transformation. The proposed grid repairing method is built upon a solid mathematic work known as the deformation based grid generation and is presented next.

3.3 Proposed Topology Preserving Regridding Method

Let ${}^{n}\varphi = {}^{n}\phi \circ {}^{n-1}\phi \circ \cdots \circ {}^{1}\phi$ denote the composite transformation using linear interpolation as described in Section 3.1. Here each transformation ${}^{i}\phi$ is related to the corresponding incremental displacement field ${}^{i}U$ through the following relation:

$$ {}^{i}\phi(X) = X + {}^{i}U(X), \tag{3.1} $$

where X is a regular grid and ${}^{i}\phi(X)$ is a deformed grid $\in \mathcal{X}$. Assume $J({}^{n}\varphi) \equiv \det \nabla^{n}\varphi \geq 0$ and $J({}^{n+1}\varphi) < 0$. That is, the topology of the deformed template $T^{(n)}(X) = T^{c}({}^{n}\varphi(X))$ is preserved but not after concatenating one more incremental transformation ${}^{n+1}\phi$. When this occurs, the following steps (algorithm 1) are proposed to repair ${}^{n+1}\phi$ such that $J({}^{n+1}\hat{\varphi})$ is strictly positive, where ${}^{n+1}\hat{\varphi} = {}^{n+1}\hat{\phi} \circ {}^{n}\phi \circ \cdots \circ {}^{1}\phi$ and ${}^{n+1}\hat{\phi}$ denotes the incremental transformation after repairing from ${}^{n+1}\phi$. In this section, we follow the notations used in chapter 2.

Algorithm 1 Proposed grid repairing method

```
1: Obtain the Jacobian determinant of the current total transformation $J({}^{n}\varphi)$
2: Find the monitor function $f_1$ and curl function $f_2$ that results in the current
   incremental displacement field ${}^{n+1}\phi$.
3: Set the modified monitor function $\hat{f}_1$ as:
       $\hat{f}_1 = \max(\frac{J_{\min}}{J({}^{n}\varphi)}, f_1),$        (3.2)
4: Construct the repaired incremental transformation ${}^{n+1}\hat{\phi}$ from $\hat{f}_1$ and $f_2$.
```

In step 3, $J_{\min}$ is the pre-set lower bound of the allowed Jacobian determininant of the displacement field and the operator is preformed element-wise. The constructing of the repaired incremental transformation ${}^{n+1}\hat{\phi}$ from $\hat{f}_1$ and f_2 in step 4 is to use the deformation based grid generation method as described in section 2.4.

Using the steps outlined above, the repaired total transformation will meet the requirement set on its Jacobin determinant. As long as $J_{\min}$ is strictly positive, grid folding is prohibited. The underlying idea is to increase the values of the monitor function f_1 at the grid points where negative Jacobian determinant values may result after concatenation of the incremental transformation with the current total transformation.

In practice, the minimum Jacobian determinan value $J_{\min}$ should not be too small. This is because after repairing, $J({}^{n+1}\hat{\varphi})$ may still be slightly less than the desired value $J_{\min}$ due to the limitation of the numerical procedures adopted. Therefore, by our experiments, it is better to repair the incremental displacement field to a slightly higher threshold. In many applications, setting both a lower bound as well as an upper bound can improve the robustness of the registration algorithm. Therefore, steps above are modified accordingly as shown in Algorithm 2.

In algorithm 2, λ_{up} is a parameter equal to or slightly greater than 1 and λ_{down} is equal to or slightly less than 1. Seeking the monitor function f_1 and curl function f_2 that will result in a known mapping as required in step 2 is in general a tough task. However, with the restriction that ${}^{n+1}\phi$ is close to an identity transformation, it is possible to devise an efficient numerical scheme to accomplish it. Therefore, maximum regridding frequency is required for grid repairing method. This is discussed in the next section.

Algorithm 2 Modified proposed grid repairing method

```
Compute the Jacobian determinant $J({}^{n}\varphi)$ which is greater than $J_{\min} > 0$ and less than $J_{\max} > J_{\min}$
Find the monitor function $f_1$ and curl function $f_2$ that results in the current incremental displacement field ${}^{n+1}\phi$.
If $\min(J({}^{n+1}\varphi)) < J_{\min}$, modify the monitor function $f_1$ by
    $\hat{f}_1 = \max(\frac{J_{\min} \cdot \lambda_{up}}{J({}^{n}\varphi)}, f_1),$    (3.3)
If $\max(J({}^{n+1}\varphi)) > J_{\max}$, modify the monitor function $f_1$ by
    $\hat{f}_1 = \min(\frac{J_{\max} \cdot \lambda_{down}}{J({}^{n}\varphi)}, f_1),$    (3.4)
Construct the repaired incremental transformation ${}^{n+1}\hat{\phi}$ from $\hat{f}_1$ and $f_2$.
```

3.4 Numerical Schemes

Numerical solvers to solve the div-curl system (Eq. (2.17)) and the time-varying velocity vector field (Eq. (2.16)) are required in the proposed grid repairing scheme. The detailed implementations of these numerical solvers used in this dissertation are described in this section. And the numerical schemes are described in Section 3.4.1 and Section 3.4.2.

3.4.1 Div-curl Solver

A div-curl system can be decoupled into d-dimensional ($d = 2$ or 3) Poisson equations [25]. A div-curl system is given by following for 3D case.

$$\begin{aligned} \operatorname{div} U &= \frac{\partial U^x}{\partial x} + \frac{\partial U^y}{\partial y} + \frac{\partial U^z}{\partial z} = f^1 \\ \operatorname{curl}_x U &= \frac{\partial U^z}{\partial y} - \frac{\partial U^y}{\partial z} = f^2 \\ \operatorname{curl}_y U &= \frac{\partial U^x}{\partial z} - \frac{\partial U^z}{\partial x} = f^3 \\ \operatorname{curl}_z U &= \frac{\partial U^y}{\partial x} - \frac{\partial U^x}{\partial y} = f^4 \end{aligned} \tag{3.5}$$

where f^1 is the scalar monitor function f, and $f^2, f^3,$ and f^4 are the three components of the curl function g. Assume 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, the following three Poisson equations are obtained,

$$\begin{aligned} \Delta U^x &= f^1_x + f^3_z - f^4_y \equiv F^1 \\ \Delta U^y &= f^1_y + f^4_x - f^2_z \equiv F^2 \\ \Delta U^z &= f^1_z + f^2_y - f^3_x \equiv F^3 \end{aligned} \tag{3.6}$$

where $f^i_k = \frac{\partial f^i}{\partial k}$. Popular choices for solving Poisson equation including, SOR, conjugate gradient, FFT, and multi-grid methods [26, 27]. In this dissertation, a FFT based Poisson solver is adopted and implemented in all the experiments.

3.4.2 ODE Solver

An ODE solver is required to solve for the mapping ϕ_t from V_t in Eq. (2.15). A popular choice for this purpose is Runge-Kutta methods [28, 29]. A higher time steps with higher order (> 2) is desired for Runge-Kutta method for higher accuracy. However, in our application, it leads the complexity to find the monitor function and curl function of a given mapping as required in Eq. (2.15). For our purpose, we adopt one-step Euler's method which is a simplest Runge-Kutta method as our ODE solver. Though this numerical scheme may seem crude, it works very well if the transformation ϕ is close to an identity mapping, or equivalently, the velocity vector field V_t is close to zero. In our proposed method in section 3.3, it is not difficult to make each incremental mapping ϕ quite small enough to warrant this requirement. The specific way to achieve this depends on the specific registration algorithm in consideration.

3.4.3 Curl Function

Given a mapping ϕ, it is straightforward to compute the corresponding monitor function f_1, which is just equal to the Jacobian determinant of ϕ according to the property given in Eq.(2.12). However, to find the corresponding curl function is not straightforward, since the curl function defines the curl of the intermediate vector filed η rather than the final mapping ϕ, and they are related through the ODE given by Eq.(2.15). As mentioned previously, in the simplest case where the ODE is solved by using the one-step Euler's method, η and ϕ can be related by

$$\phi(X) \;=\; \phi_1(X) \;=\; \phi_0(X) + V_1(\phi_0(X)) \;=\; Id(X) + \eta(Id(X)) \;=\; X + \eta(X) \quad (3.7)$$

where Id denotes the identity mapping. Consequently in this case, the intermediate vector field is nothing but the displacement field itself. Following Eq. ss (2.17), the curl function can be calculated as

$$f_2 \equiv curl\eta = curl(\phi - Id) \quad (3.8)$$

3.5 Integration of Regridding

The regridding scheme can be incorporated into most nonrigid image registration algorithms as long as the registration algorithm solves displacement field iteratively. Here we integrate the proposed method in elastic, fluid, diffusion, curvature and demons registration algorithms as reviewed in chapter 2. The general flow chart for the itergration of regridding is given in Fig. 3.4. Initially $i = 1$ and the template $T^{(0)}$ is the original template T. Total transformation is denoted by ${}^{(i)}\varphi$ and incremental transformation ${}^{(k)}\phi$. The initial totoal transformation ${}^{(i)}\varphi = I$, where I is the identity transformation. After one iteration of registration, a new incremental transformation ${}^{(k)}\phi$ is produced by ${}^{(k)}\phi = I + {}^{(k)}u$ where ${}^{(k)}u$ is the displacement calculated by the employed nonrigid image registration algorithm. If the minmum Jacobian deterninant of ${}^{(k)}\phi$ is less than a preset propagation threshold, regridding is performed. A concatenated transformation is obtained by ${}^{(i)}\varphi = {}^{(i-1)}\varphi \circ {}^{(k)}\phi$ where ${}^{(i-1)}\varphi$ is composited with ${}^{(k)}\phi$ by using linear interpolation. k is reset to 1, the displacement ${}^{(k)}u$ is reset to 0. A new templete $T^{(i)}$ is resampled from $T^{(0)}$ by using the transformation ${}^{(i)}\varphi$. Finally, i is increased by one and registration re-starts unitl a pre-determined termination criterian is satisfied. To include the grid reparing method described in section 3.3, the final block shown in Fig. 3.4 is replaced by Fig. 3.5 The method is described as following. Grid repairing is activated whenever the minimum Jacobian determinant of ${}^{(i)}\varphi$ is less than a preset threshold or the maximum Jacobian determinant of ${}^{(i)}\varphi$ is greater than a preset threshold. ${}^{(i)}\hat{\varphi}$ is obtained by composite ${}^{(k)}\hat{\phi}$ with ${}^{(i-1)}\varphi$ where ${}^{(k)}\hat{\phi}$ is the repaired incremental transformation from ${}^{(k)}\phi$. Otherwise, ${}^{(i)}\hat{\varphi} = {}^{(i)}\varphi$. Using this method, it is necessary to maximize the regridding frequency due to the numerical accuracy of grid generation method as detailed in section 3.3.

To illustrate how exactly the proposed regridding scheme can be incorporated into existing nonrigid image registration algorithms, we demonstrate it using the methods reviewed in chapter 2. And the detail algorithm is given as in algorithm 3,

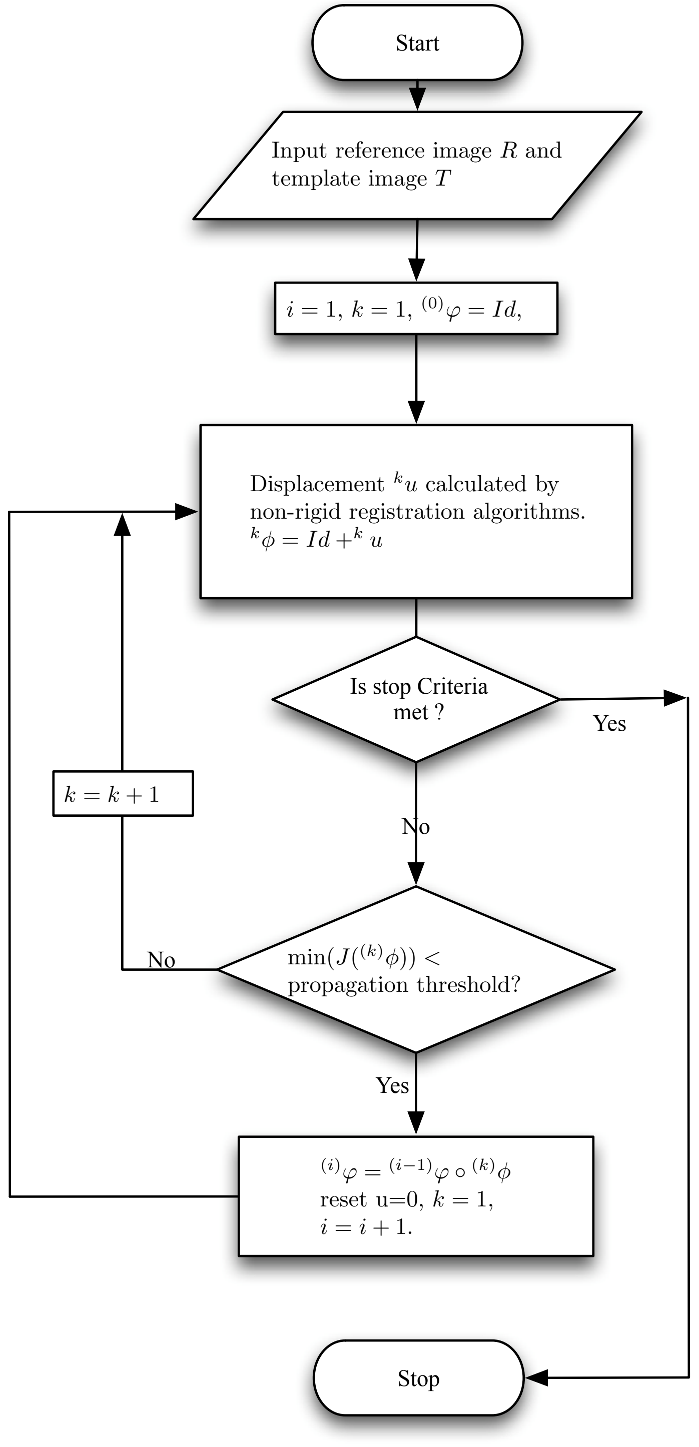
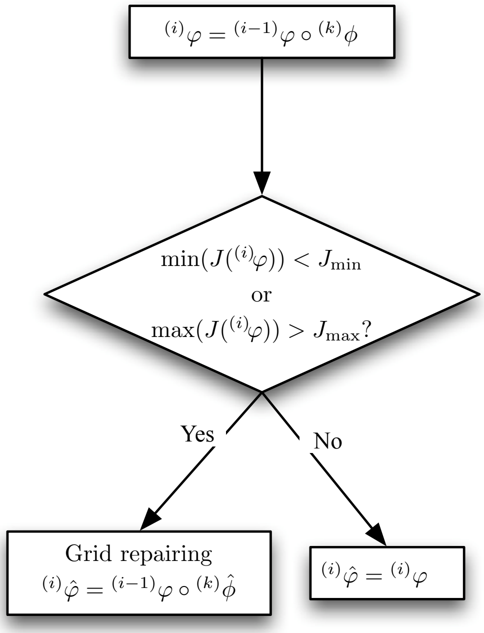

Start
Input reference image R and template image T
$i = 1$, $k = 1$, ${}^{(0)}\varphi = Id$,
Displacement ${}^{k}u$ calculated by non-rigid registration algorithms.
${}^{k}\phi = Id +^{k} u$
Is stop Criteria met ?
Yes
No
$k = k + 1$
No
$\min(J({}^{(k)}\phi)) <$ propagation threshold?
Yes
${}^{(i)}\varphi = {}^{(i-1)}\varphi \circ {}^{(k)}\phi$
reset u=0, $k = 1$,
$i = i + 1$.
Stop

Figure 3.4. Flow chart of nonrigid image registration incorporated with the regridding method.



$^{(i)}\varphi = {}^{(i-1)}\varphi \circ {}^{(k)}\phi$
$\min(J(^{(i)}\varphi)) < J_{\min}$
or
$\max(J(^{(i)}\varphi)) > J_{\max}$?
Yes
No
Grid repairing
$^{(i)}\hat{\varphi} = {}^{(i-1)}\varphi \circ {}^{(k)}\hat{\phi}$
$^{(i)}\hat{\varphi} = {}^{(i)}\varphi$

Figure 3.5. Flow chart of grid repairing.

3.5.1 Dynamic Step Size Adjustment

Setting the external force to be the termination criterion of the registration is suggested in [11]. In addition, fixed step size usually results low convergence rate. Algorithm 4 is then proposed to improve the performance by adjusting step size dynamically. To allow the step size to be adjusted dynamically in algorithms 3, we simply multiply it by a constant $\tau_{up} > 1$ if the similarity measure improves; otherwise, it is multiplied by another constant $\tau_{down} < 1$. Algorithms are terminated if either the maximum number of iterations is reached, or the step size τ is less than a predefined threshold τ_{min}. The details of the modification are given in algorithm 4.

Algorithm 3 Registration with proposed regridding method 1

```
1: Let $i$=0, $T^{(0)}(X) = T(X)$ , and ${}^{0}u(X) = 0$.
2: Compute the external force.
3: If the external force is below than a threshold or the maximum number of iterations is reached, then STOP.
4: Solve the displacement field ${}^{i+1}U(X)$ by different registration algorithms. Assuming that ${}^{i+1}U(\tilde{x}) = 0$, $\tilde{x}$ is a grid point on the border.
5: Scale the (i+1)$^{th}$ displacement field using ${}^{i+1}U(X) = tstep \cdot \frac{{}^{i+1}U(X)}{\gamma}$, where $\gamma = \max||{}^{i+1}U(X)||$.
6: Concatenate the (i+1)$^{th}$ displacement field ${}^{i+1}U(X)$ into the $i$th total displacement field ${}^{i}u(X)$ to produce the (i+1)$^{th}$ total displacement field ${}^{i+1}u(X)$ using ${}^{i+1}u(X) = {}^{i+1}U(X) + {}^{i}u(X + {}^{i+1}U(X))$ . Calculate the Jacobian determinant of the concatenated transformation $J(X + {}^{i+1}u(X))$. If $\min(J(X + {}^{i+1}u(X))) < J_{min}$ or $\max(J(X + {}^{i+1}u(X))) > J_{max}$, repair the (i+1)$^{th}$ incremental displacement field ${}^{i+1}U(X)$ to produce ${}^{i+1}\tilde{U}(X)$ using algorithm 2 and set ${}^{i+1}u(X) = {}^{i+1}U(X) + {}^{i}u(X + {}^{i+1}\tilde{U}(X))$.
7: Resample the original template to obtain the (i+1)$^{th}$ deformed template using $T^{(i+1)}(X) = T^{(0)}(X + {}^{i+1}u(X))$. Set $i$ = $i$ + 1 and go to step 2
```

Algorithm 4 Registration with proposed regridding method 2

```
1: Let $i$=0, $T^{(0)}(X) = T(X)$ , ${}^{0}u(X) = 0$, and initialize the step size $\tau = 0.5$.
2: If $\tau < \tau_{min}$ or the maximum number of iterations is reached, then STOP.
3: Solve the displacement field ${}^{i+1}U(X)$ by different registration algorithms. Assuming that ${}^{i+1}U(\tilde{x}) = 0$, $\tilde{x}$ is a grid point on the border.
4: Scale the (i+1)$^{th}$ displacement field using ${}^{i+1}U(X) = \tau \cdot \frac{{}^{i+1}U(X)}{\gamma}$, where $\gamma = \max||{}^{i+1}U(X)||$.
5: Concatenate the (i+1)$^{th}$ displacement field ${}^{i+1}U(X)$ into the $i$th total displacement field ${}^{i}u(X)$ to produce the (i+1)$^{th}$ total displacement field ${}^{i+1}u(X)$ using ${}^{i+1}u(X) = {}^{i+1}U(X) + {}^{i}u(X + {}^{i+1}U(X))$ . Calculate the Jacobian determinant of the concatenated transformation $J(X + {}^{i+1}u(X))$. If $\min(J(X + {}^{i+1}u(X))) < J_{min}$ or $\max(J(X + {}^{i+1}u(X))) > J_{max}$, calculate the Jacobian determinant of the incremental transformation $J(X + {}^{i+1}U(X))$. If $\min(J(X + {}^{i+1}U(X))\ ) > 0.97$, repair the (i+1)$^{th}$ incremental displacement field ${}^{i+1}U(X)$ to produce ${}^{i+1}\tilde{U}(X)$ using algorithm 2 and set ${}^{i+1}u(X) = {}^{i+1}U(X) + {}^{i}u(X + {}^{i+1}\tilde{U}(X))$.
6: Resample the original template to obtain the (i+1)$^{th}$ deformed template using $T^{(i+1)}(X) = T^{(0)}(X + {}^{i+1}u(X))$. Update the similarity measure. If the similarity measure improves, $\tau = \tau \cdot \tau_{up}$, otherwise, $\tau = \tau \cdot \tau_{down}$. Let $i = i + 1$ and go to step 2.
```

CHAPTER 4

EXPERIMENTAL RESULTS

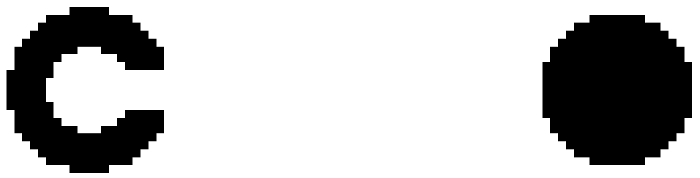
In this chapter we show registration experiments by integrating our proposed method in modified version of elastic (as described in section 2.1.1), fluid, diffusion, curvature and demons registration algorithms. Comparison are made to the quality and robustness of the original method, maximum regridding frequency and our proposed method. All the experiments are conducted on the Max OS X 10.5.5 Platform with 2 Xeon 2.66 GHz and 5 GB ram. Most parts are implemented in MATLAB, and some critical computational parts are implemented by using the mex in C and C++.

4.1 Experiment One: • to C

The purpose of this experiment is to test the ability of different regridding methods applied on each registration algorithms in this dissertation for a large deformatin application.

4.1.1 Experimental Design

We start with a widely used exmaple, given the two images depicted in Fig. 4.1, both are 33 by 33 pixels binary images. Three schemes for each registration algorithm are applied: original algorithm, maximum regridding frequency, and regridding with our proposed method. All registrations started with $u^0 = 0$ and were stopped as soon as $\det(J) < 0$ or the SSD is not improved for 6 iterations.



(a)

(b)

Figure 4.1. 'C'-experiment: 33 × 33 (a)binary image C (b)binary image •.

4.1.2 Elastic Registration: • to C

The Lame constants of are set to be $\mu = 30$, $\lambda = 50$ for the elastic registration. In Fig. 4.2, we show intermediate results for $k = 20(20)220, 278$ of the modified elastic registration (Section 2.1.1). The modified elastic registration stopped at $k = 278$ when the Jacobian determinant of the transformation is negative as shown in Fig 4.3b. We zoom in the transformation of the final grid and the folded grid can be found in Fig. 4.2o. The maximum regridding frequency of the original elastic registration is shown in Fig. 4.4. Obviously, SSD can be improved from 7000 to 20, and the minimum of $\det(J)$ is from 1 to 0.07. Grid is not folded as shown in Fig. 4.4p and the $\det(J)$ does not go below than 0 as shown in Fig. 4.5b. Here, we first demonstrate that the large-deformation application can be done by applying maximum regridding frequency. To show the effects of the proposed grid repairing scheme on the original elastic registration, Fig. 4.6 displays the results from Algorithm 4 with $J_{min} = 0.3$,

$J_{max} = \infty$, $\tau_{up} = 1$, and $\tau_{down} = 1$. The minimum of the Jacobian determinant value is successfully maintained around 0.3 (J_{min}) and with final value 0.299492. Not only the grid is not folded, but also the Jacobian determinant is able to be controlled. The effectiveness of the proposed grid repairing mechanism is therefore evidenced.

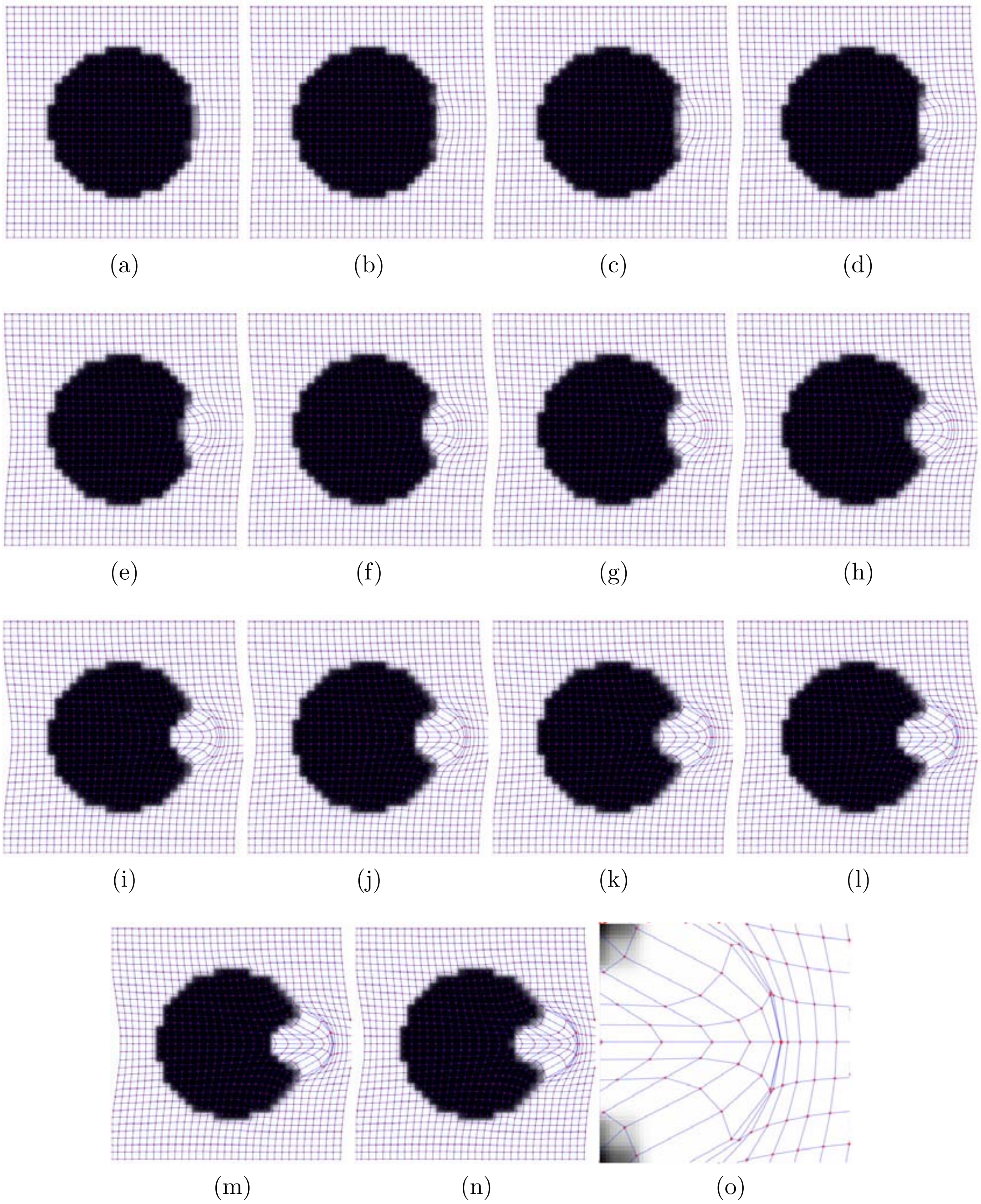
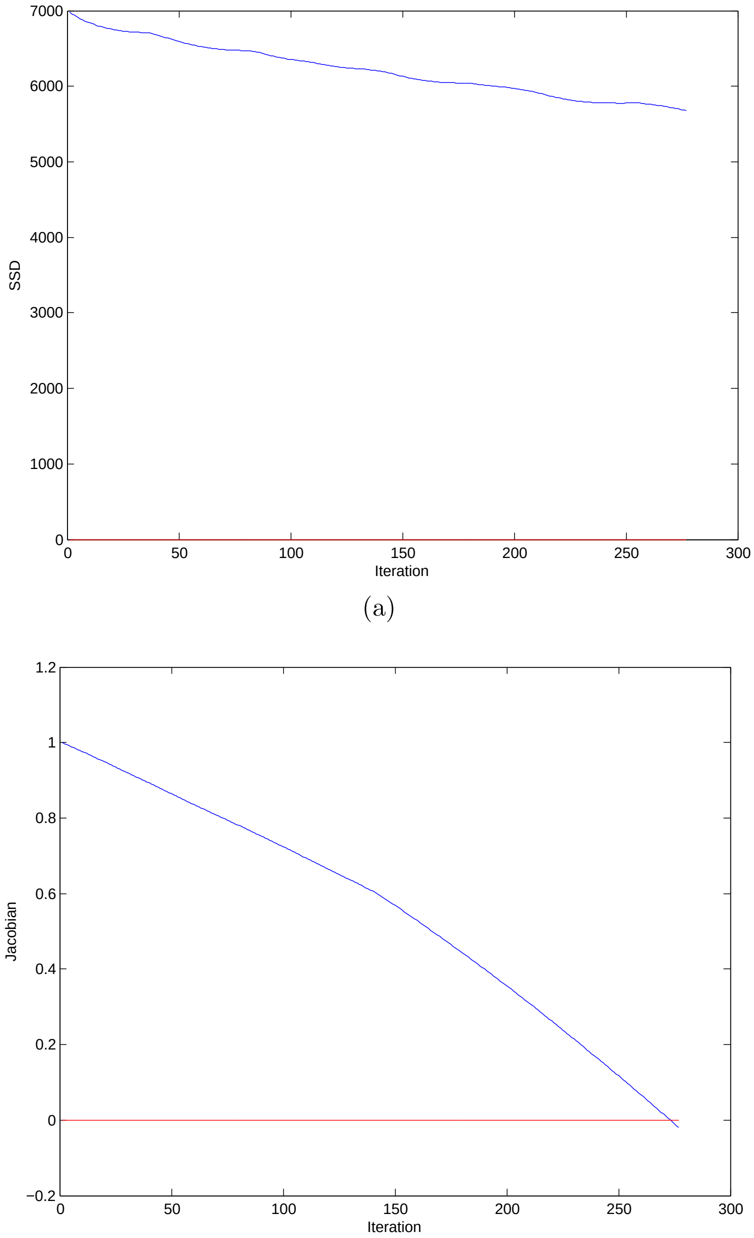


Figure 4.2. • to C results of the modified version of the original elastic registration, $\mu = 30$, $\lambda = 50$; (a)-(n) intermediate results for k=20(20)220, 278 (o) zoom in the folded grid.



(a)

(b)

Figure 4.3. ● to C results of the modified version of the original elastic registration, $\mu = 30$, $\lambda = 50$; (a) SSD and (b) Jacobian measurement for 287 iterations (SSD=5681.4, J=-0.02 at the 287th iteration).

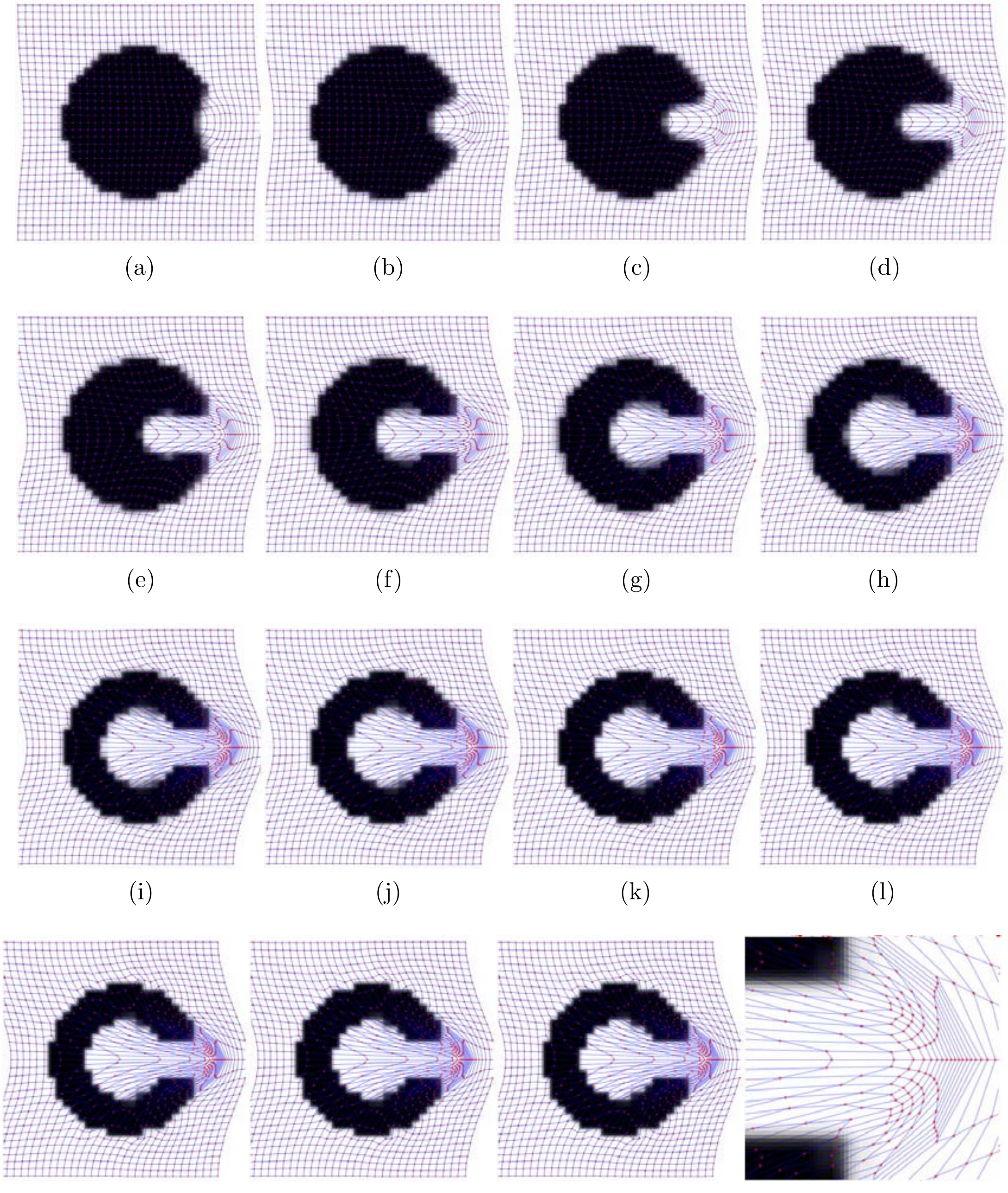
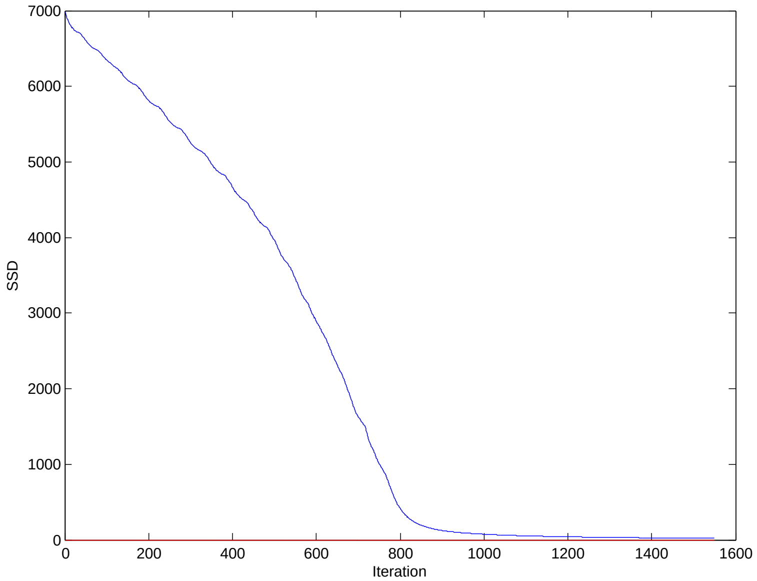
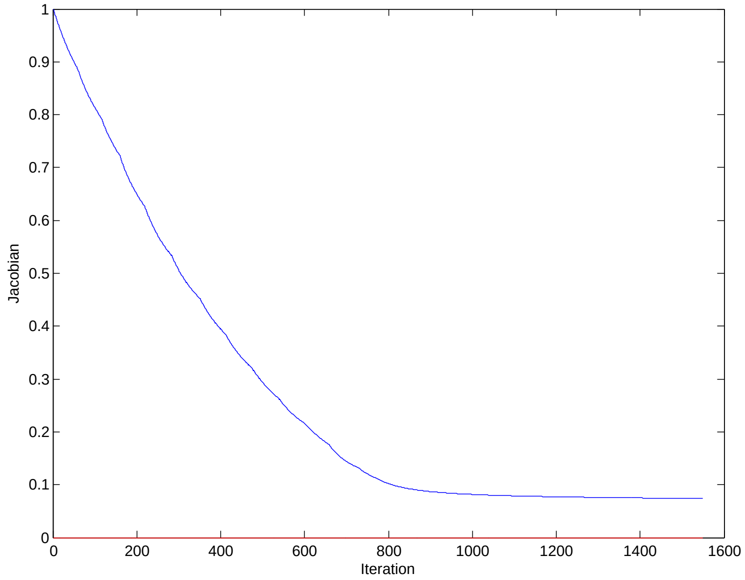


Figure 4.4. • to C of the original elastic registration with the maximum regridding frequency, $\mu = 30$, $\lambda = 50$; (a)-(o) intermediate results for k=100(100)1400,1550 (p) zoom in the grid.



(a)



(b)

Figure 4.5. • to C results of the elastic registration with the maximum regridding frequency, $\mu = 30$, $\lambda = 50$; (a) SSD and (b) Jacobian measurement for 1550 iterations (SSD=19.855499, J=0.074026245 at the 1550th iteration).

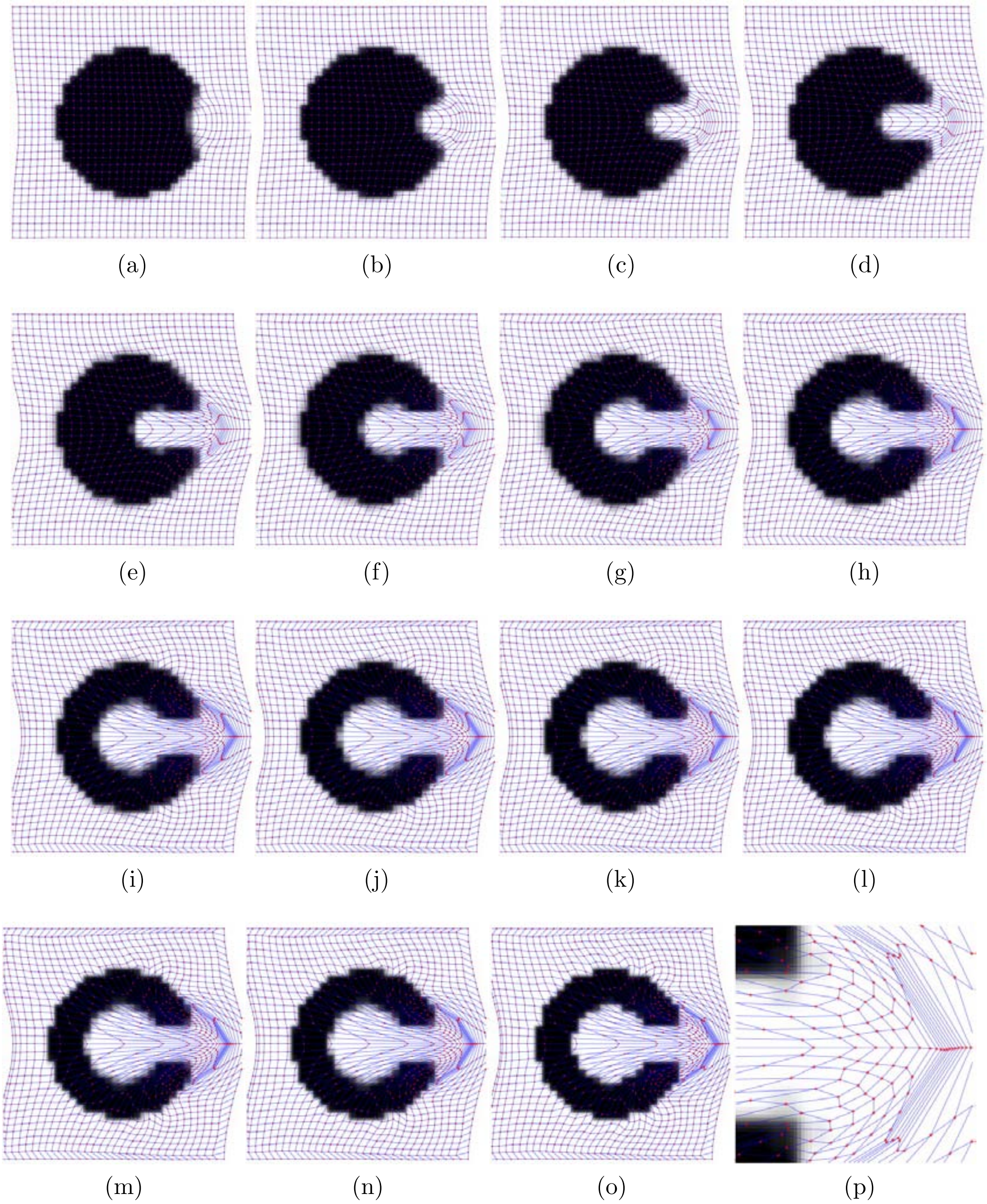
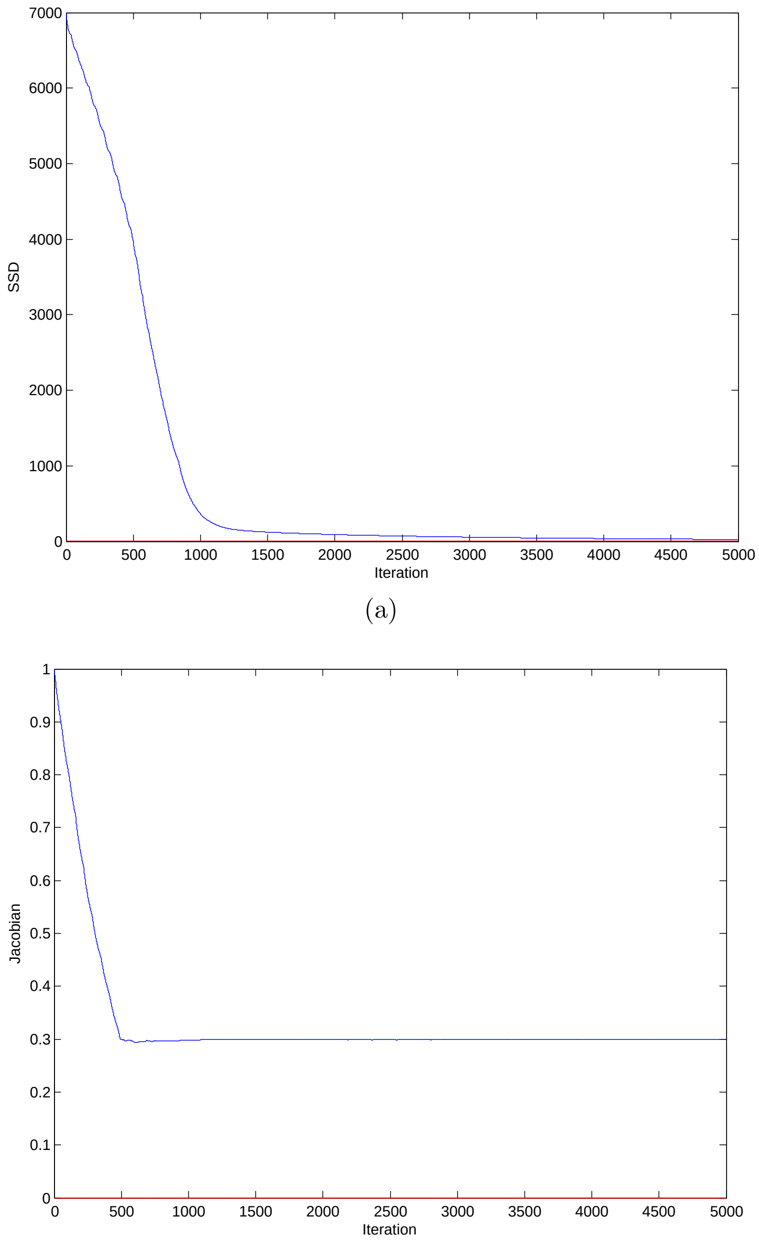


Figure 4.6. ● to C results of the elastic registration with proposed regridding method, $\mu = 30$, $\lambda = 50$; (a)-(o) intermediate results for k=100(100)1400, 5000 (p) zoom in the grid.



(a)

(b)

Figure 4.7. ● to C results of the elastic registration with proposed regridding method, $\mu = 30$, $\lambda = 50$; The Jacobian threshold is set to be 0.3, (a) SSD and (b) Jacobian measurement for 5000 iterations (SSD=25.219275 J=0.299492 at the 5000th iteration).

4.1.3 Fluid Registration: • to C

In Fig. 4.8, we show the results of using fluid registration without any regridding involved. In this example, the Lame constants are the same as the elastic registration: $\lambda = 50$ and $\mu = 30$. The transformation grid is folded at the 1050th iteration as shown in the Fig. 4.8k and Fig. 4.8l. Fig. 4.9b indicates that the Jacobian becomes negative (folding) at the 1050th iteration.

The maximum regridding frequency of the fluid registration is shown in Fig. 4.10. SSD can be improved from 7000 to 41.72, and the minimum of $\det(J)$ is from 1 to 0.077504. Grid of the transformation is not folded as shown in Fig. 4.10p and the $\det(J)$ is also positive as shown in Fig. 4.11b.

To show the effects of the proposed grid repairing scheme on the fluid registration, Fig. 4.12 displays the results from Algorithm 4 with $J_{min} = 0.3$, $J_{max} = \infty$, $\tau_{up} = 1$, and $\tau_{down} = 1$. The minimum of the Jacobian determinant value is successfully maintained around 0.3 (J_{min}) and with final value 0.299798.

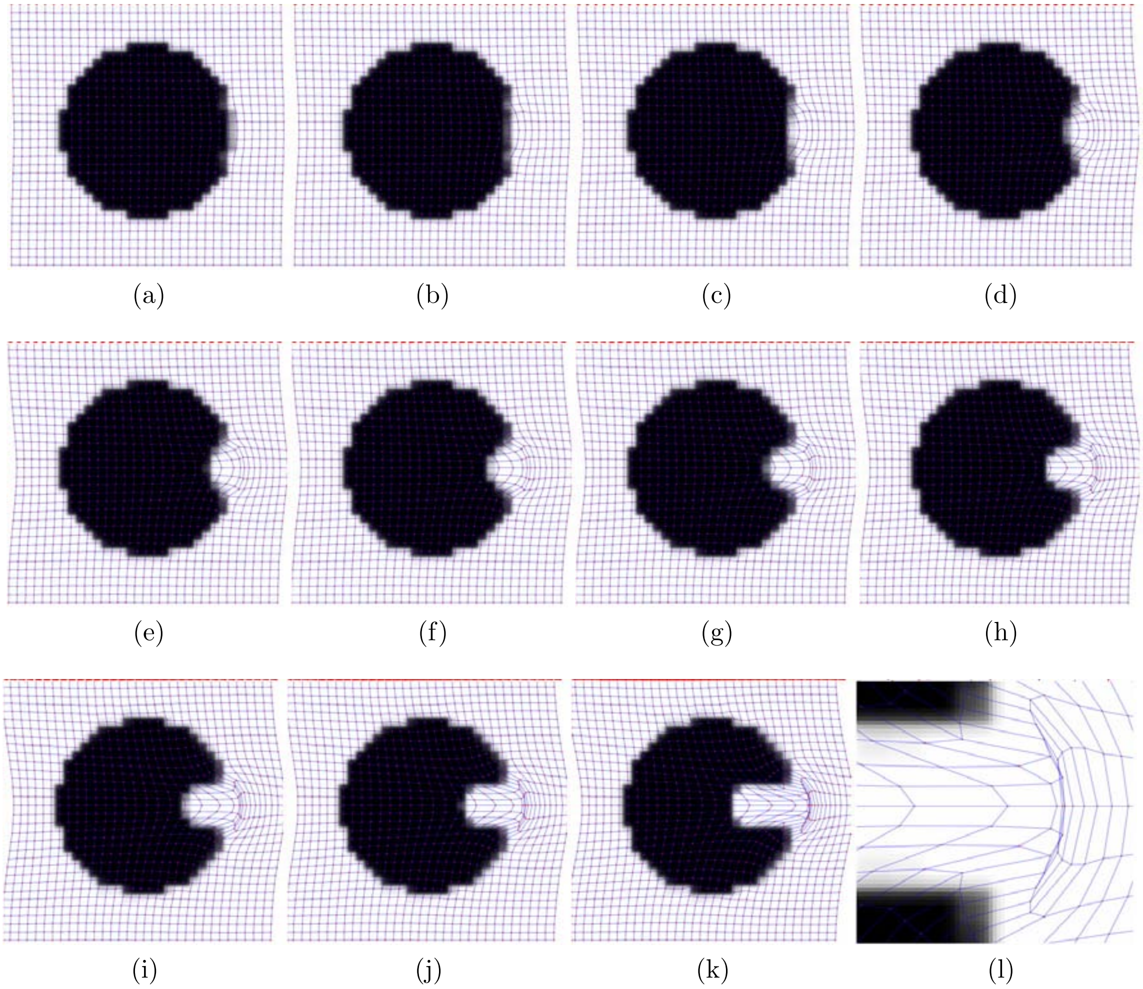
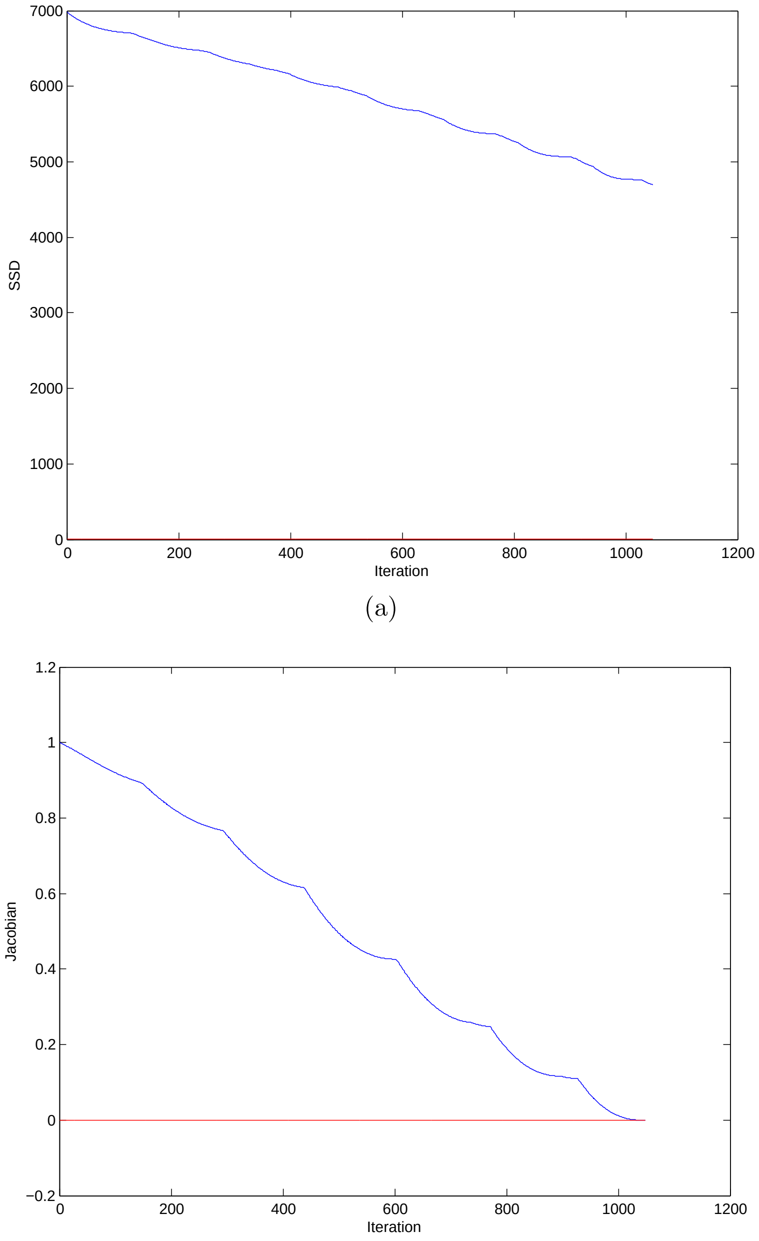


Figure 4.8. • to C results of the fluid registration without any regridding involved, $\mu = 30$, $\lambda = 50$; (a)-(k) intermediate results for k=80(80)800,1050 (l) zoom in the folded grid.



(a)

(b)

Figure 4.9. $\bullet$ to C results of the fluid registration without any regridding involved, $\mu = 30$, $\lambda = 50$; (a) SSD and (b) Jacobian measurement for 1050 iterations (SSD=4695.307740 J=-0.001168 at the 1050th iteration).

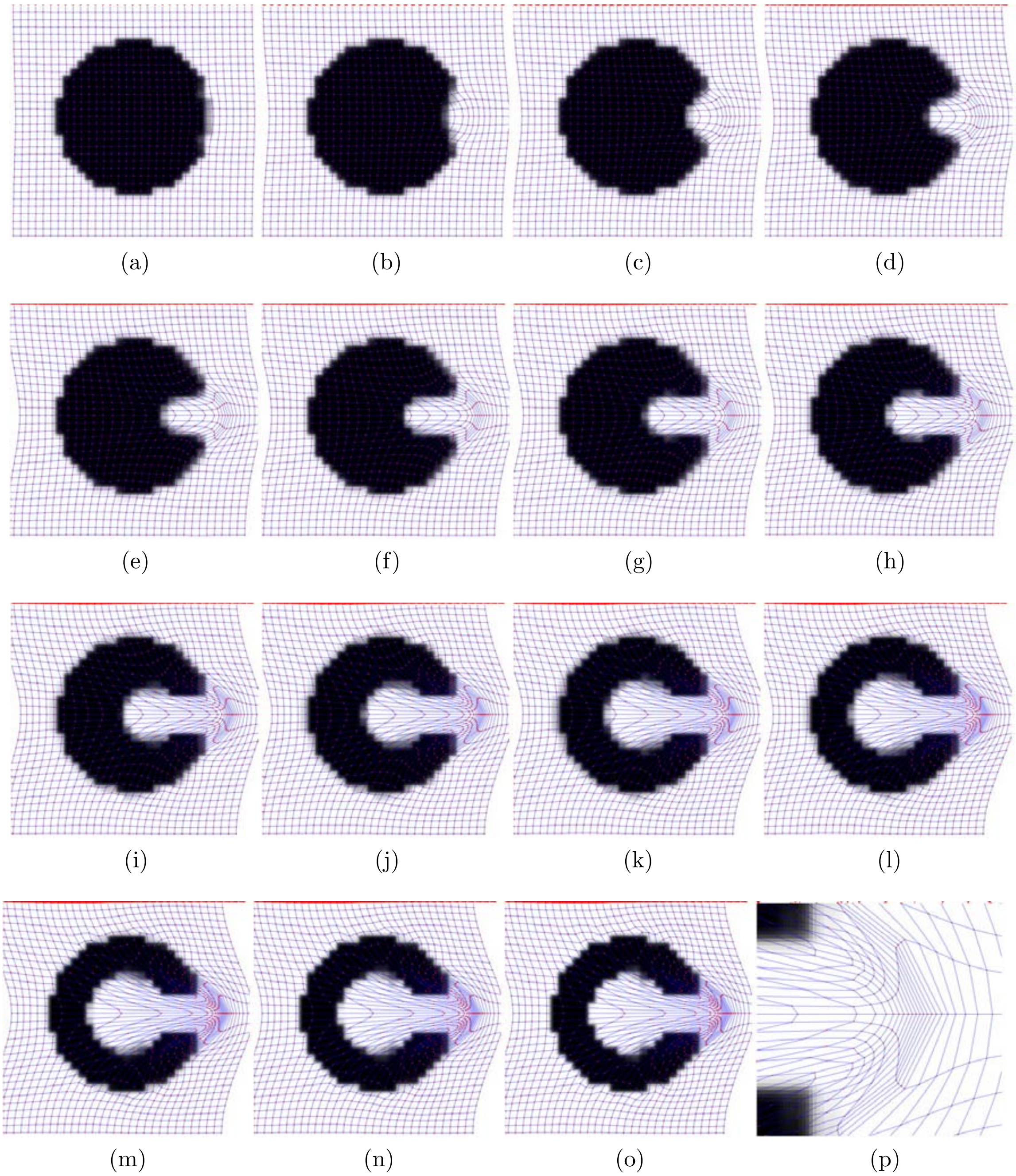
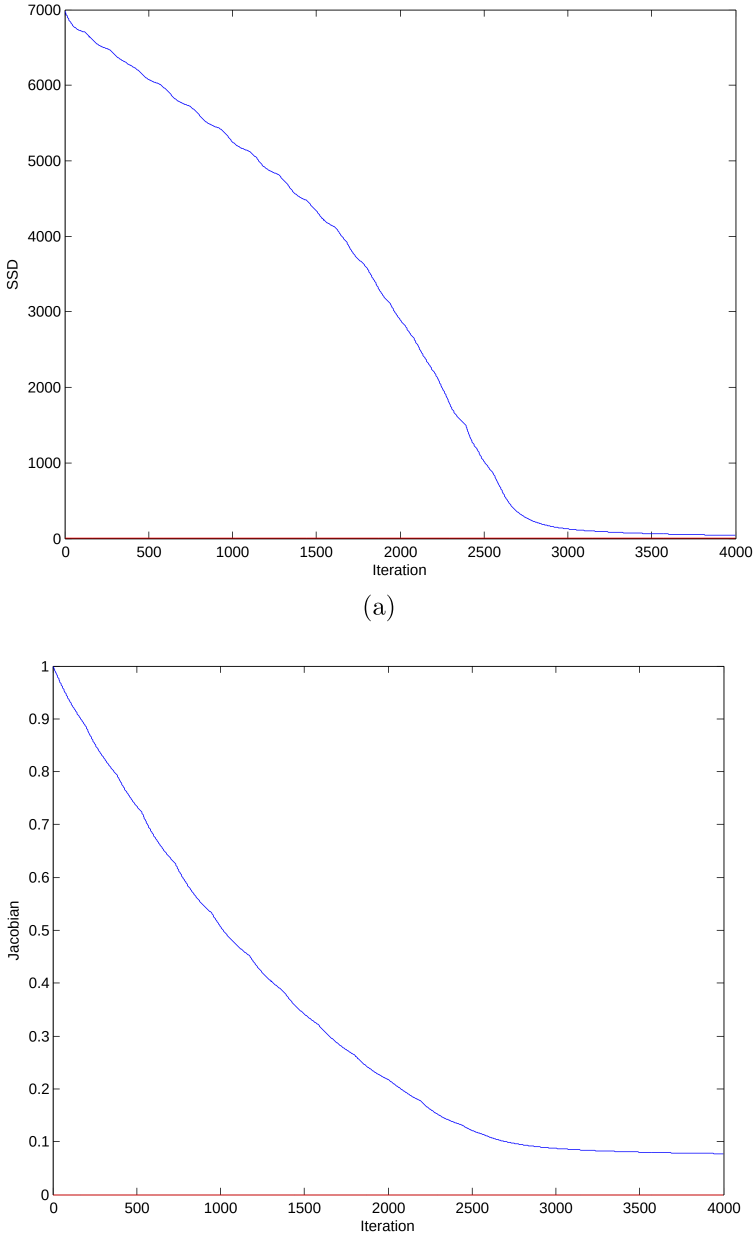


Figure 4.10. • to C results of the fluid registration with the maximum regridding frequency, $\mu = 30$, $\lambda = 50$; (a)-(o) intermediate results for k=80(240)3200,4000 (p) zoom in the grid.



(a)

(b)

Figure 4.11. • to C results of the fluid registration with the maximum frequency, $\mu = 30$, $\lambda = 50$; (a) SSD and (b) Jacobian measurement for 4000 iterations (SSD=41.720483 J=0.077504 at the 4000th iteration).

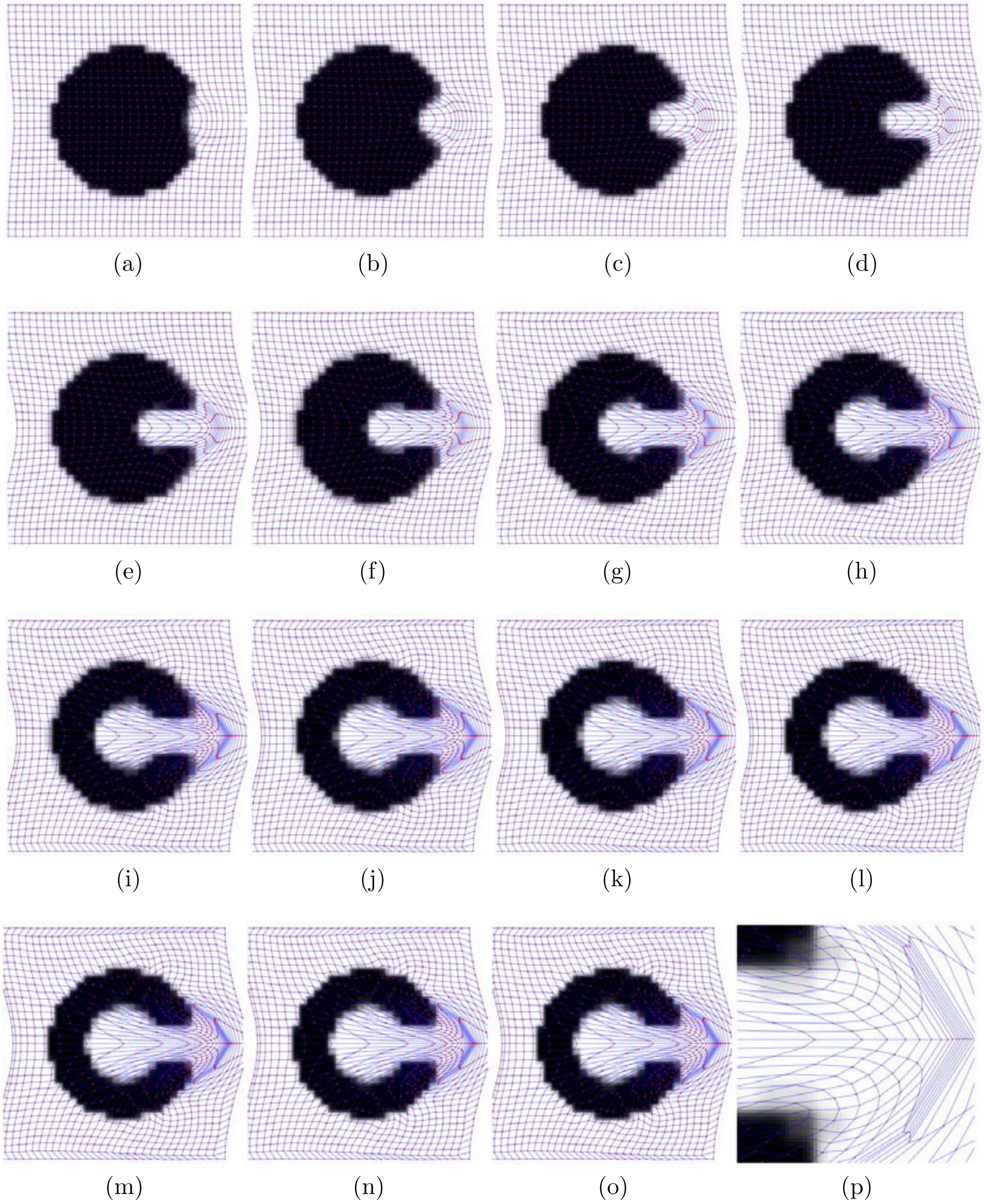
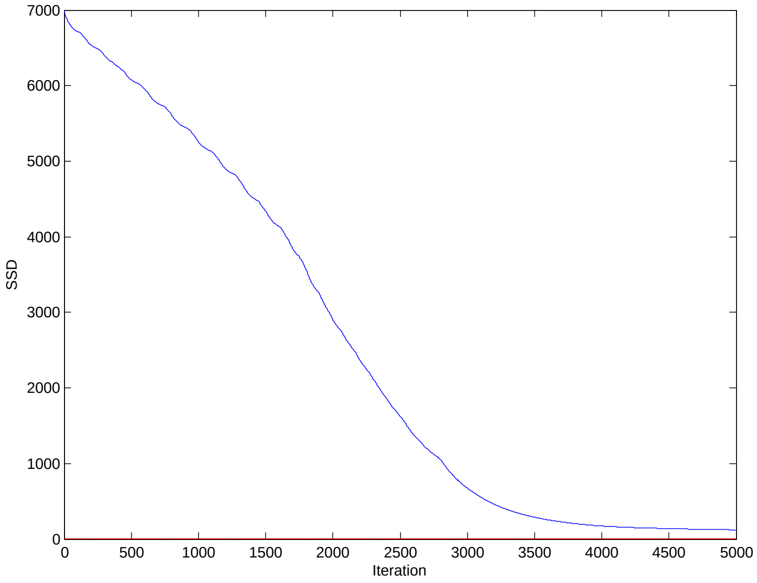
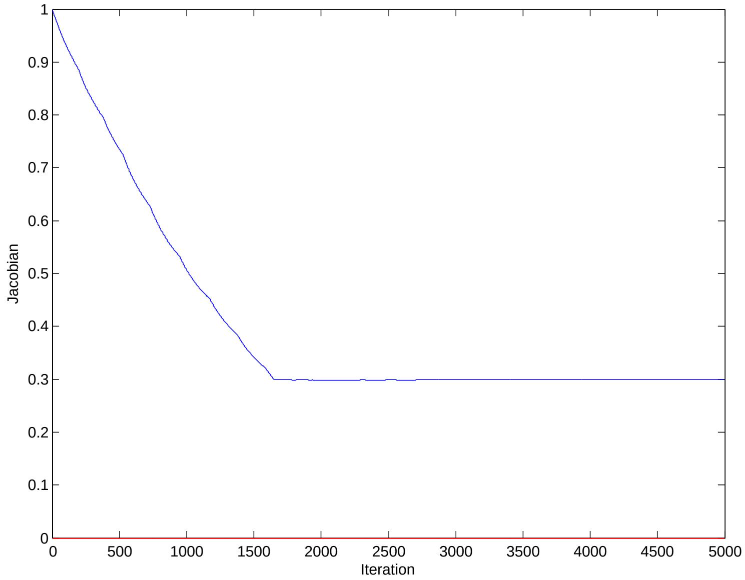


Figure 4.12. • to C results of the fluid registration with the proposed regridding method, $\mu = 30$, $\lambda = 50$; (a)-(o) intermediate results for k=300(300)4500, 5000 (p) zoom in the grid.



(a)



(b)

Figure 4.13. • to C results of the fluid registration with the proposed regridding method, $\mu = 30$, $\lambda = 50$; The minimum of Jacobian (J_{min}) is set to be 0.3, (a) SSD and (b) Jacobian measurement for 5000 iterations (SSD=122.739863 J=0.299798 at the 5000th iteraion).

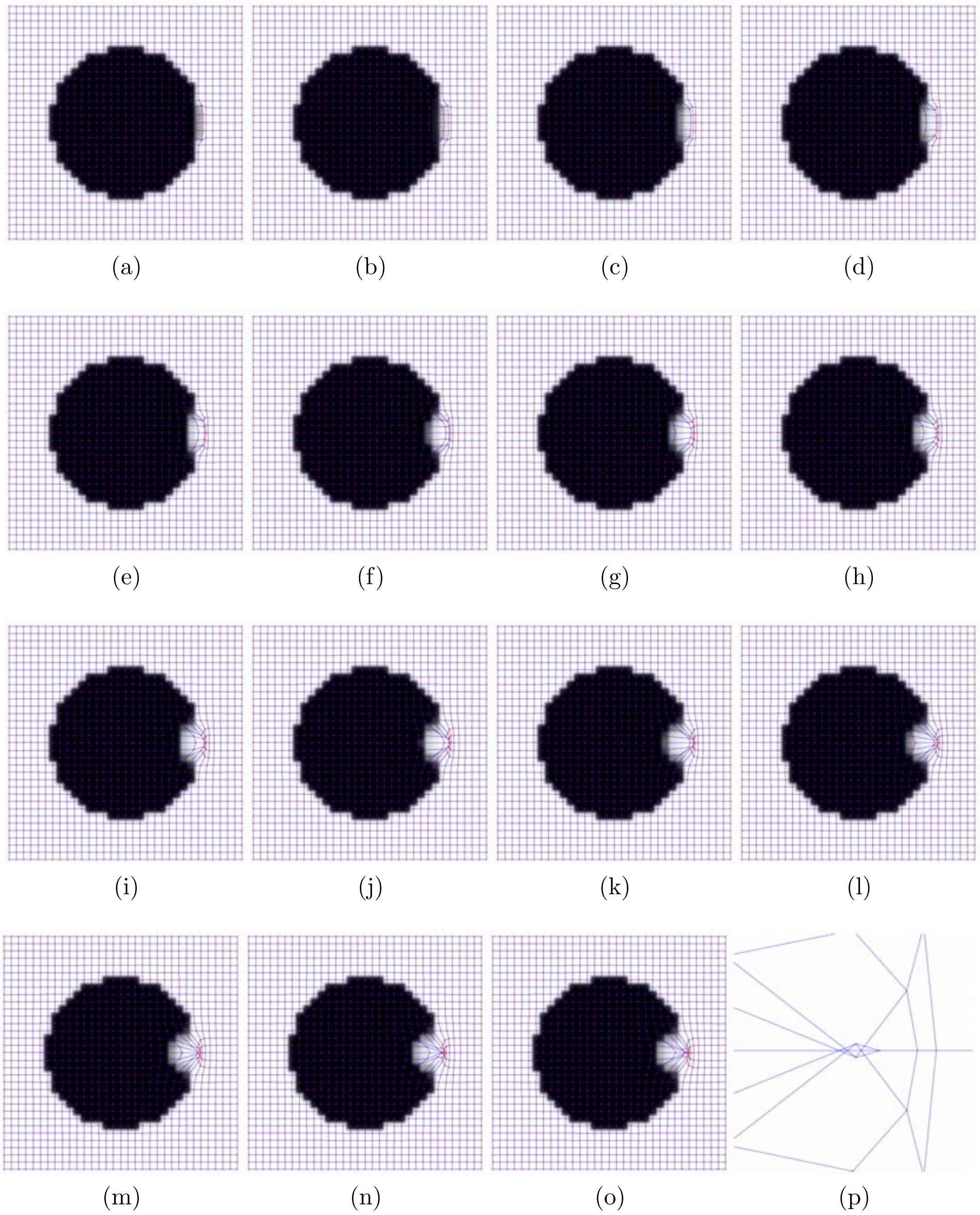
4.1.4 Diffusion Registration: • to C

In Fig. 4.14, we show the results of using the original diffusion registration. In this example, the parameters of the diffusion registration are $\alpha = 1$, $\tau = 0.01$. The results show that the original diffusion registration algorithm is not suitable for large-deformation cases.

The improvement of the diffusion registration with maximum regridding frequency is shown in Fig. 4.16. SSD can be improved from 7000 to 402.321473, but the transformation grid is folded at the 527th iteration as shown in Fig. 4.16l and Fig. 4.17b

To show the effects of the proposed grid repairing scheme on the fluid registration, Fig. 4.18 displays the results from Algorithm 4 with $J_{min} = 0.3$, $J_{max} = \infty$, $\tau_{up} = 1$, and $\tau_{down} = 1$. The minimum of the Jacobian determinant value is successfully maintained around 0.3 (J_{min}) and with final value 0.296416.

Some flutter can be observed in Fig. 4.19b due to the displacement of some iterations jump sharply according to the numerical accuracy. One can adjust the weight (α) of the regularization term to have a stronger internal force and smoother displacement. In this example, we are showing the ability of our proposed method to maintained the determinant of Jacobian around the predefined threshold for the diffusion registration method.



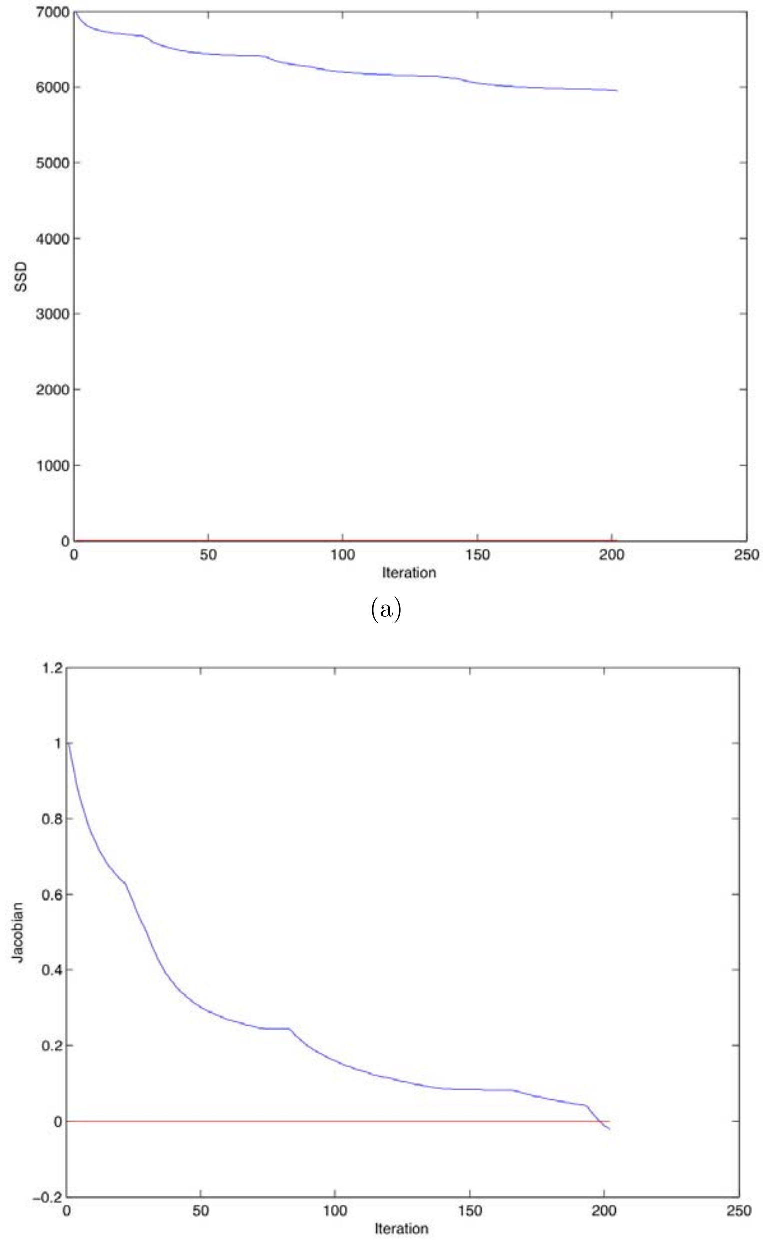
(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

Figure 4.14. • to C results of the original diffusion registration, $\alpha = 1$, $\tau = 0.01$; (a)-(o) intermediate results for k=14(14)196, 203 (p) zoom in the folded grid.



(a)

(b)

Figure 4.15. • to C of the original diffusion registration, $\alpha = 1$, $\tau = 0.01$; (a) SSD and (b) Jacobian measurement for 203 iterations (SSD=5947.925594 J=-0.021337 at the 203rd iteration).

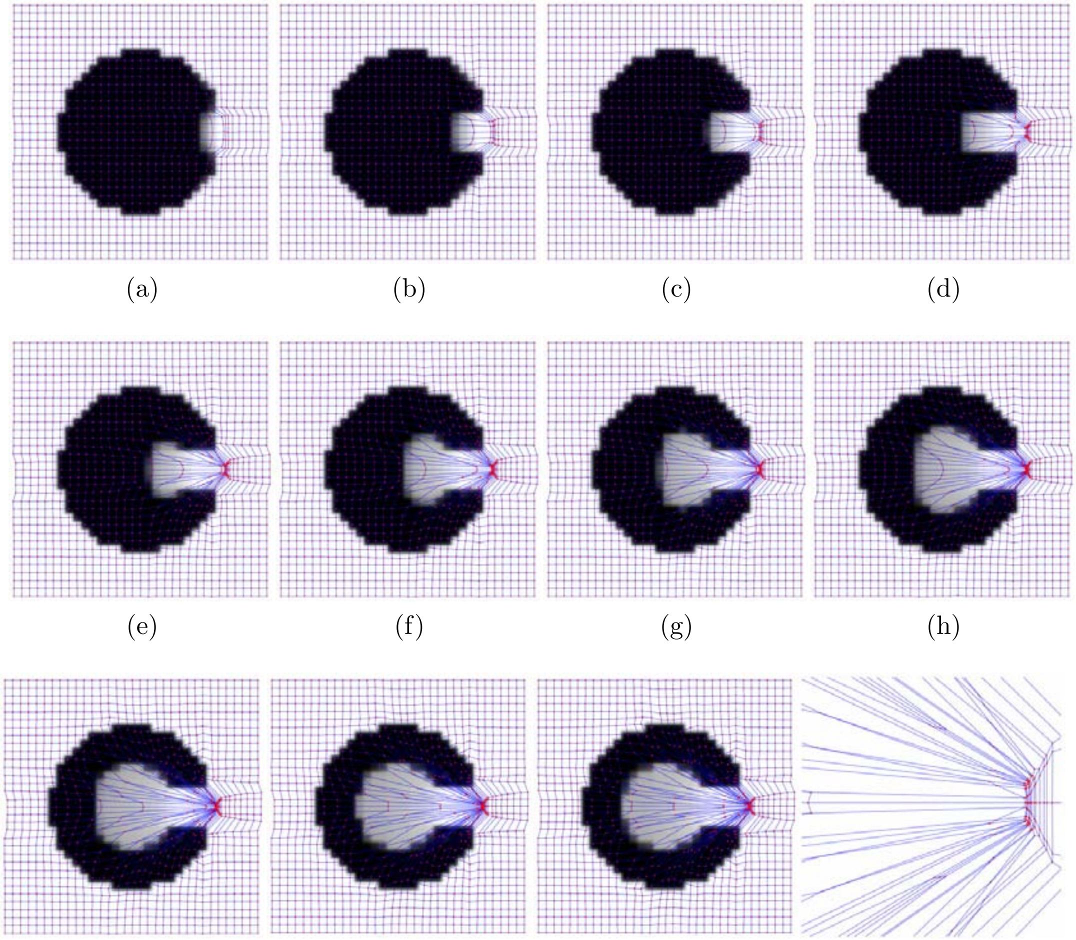
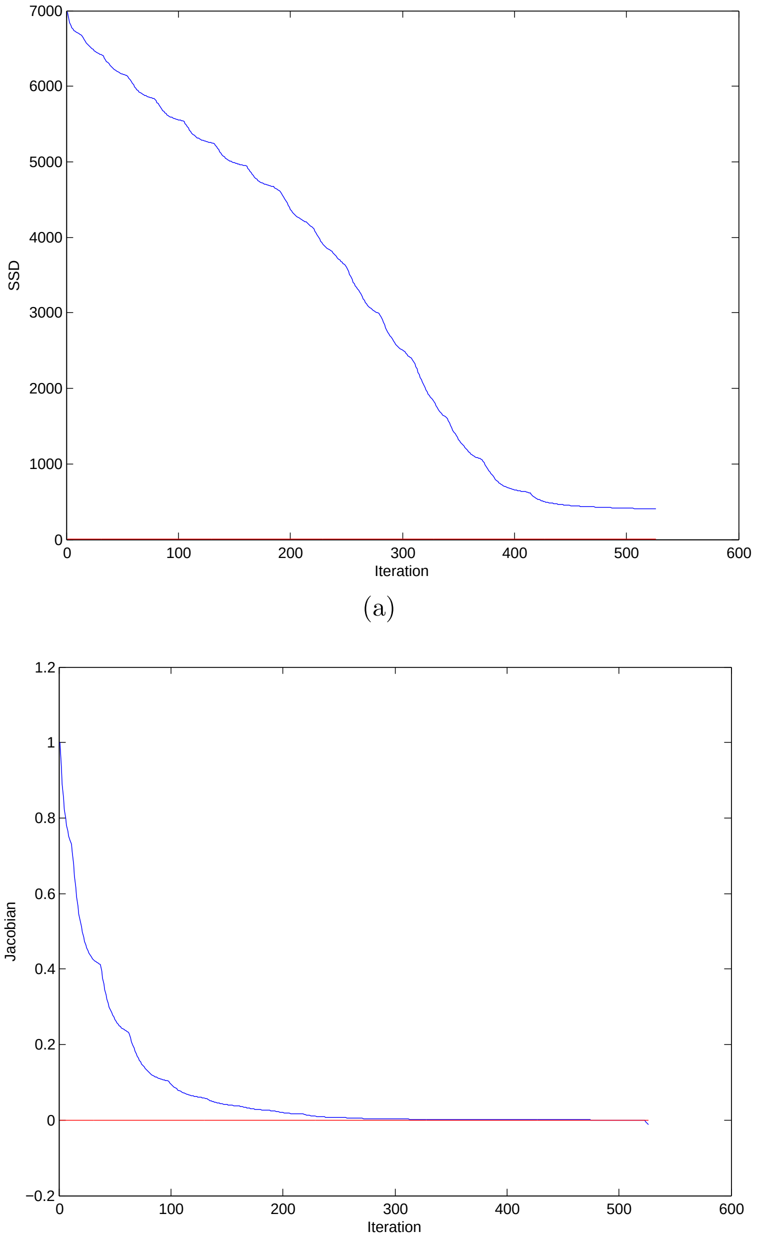


Figure 4.16. • to C results of the diffusion registration with maximum regridding frequency, $\alpha = 1$, $\tau = 0.01$; (a)-(k) intermediate results for k=45(45)450, 527 (l) zoom in the grid.



(a)

(b)

Figure 4.17. • to C results of the diffusion registration with maximum regridding frequency, $\alpha = 1$, $\tau = 0.01$; (a) SSD and (b) Jacobian measurement for 527 iterations (SSD=402.321473 J=-0.011998 at the 527th iteration).

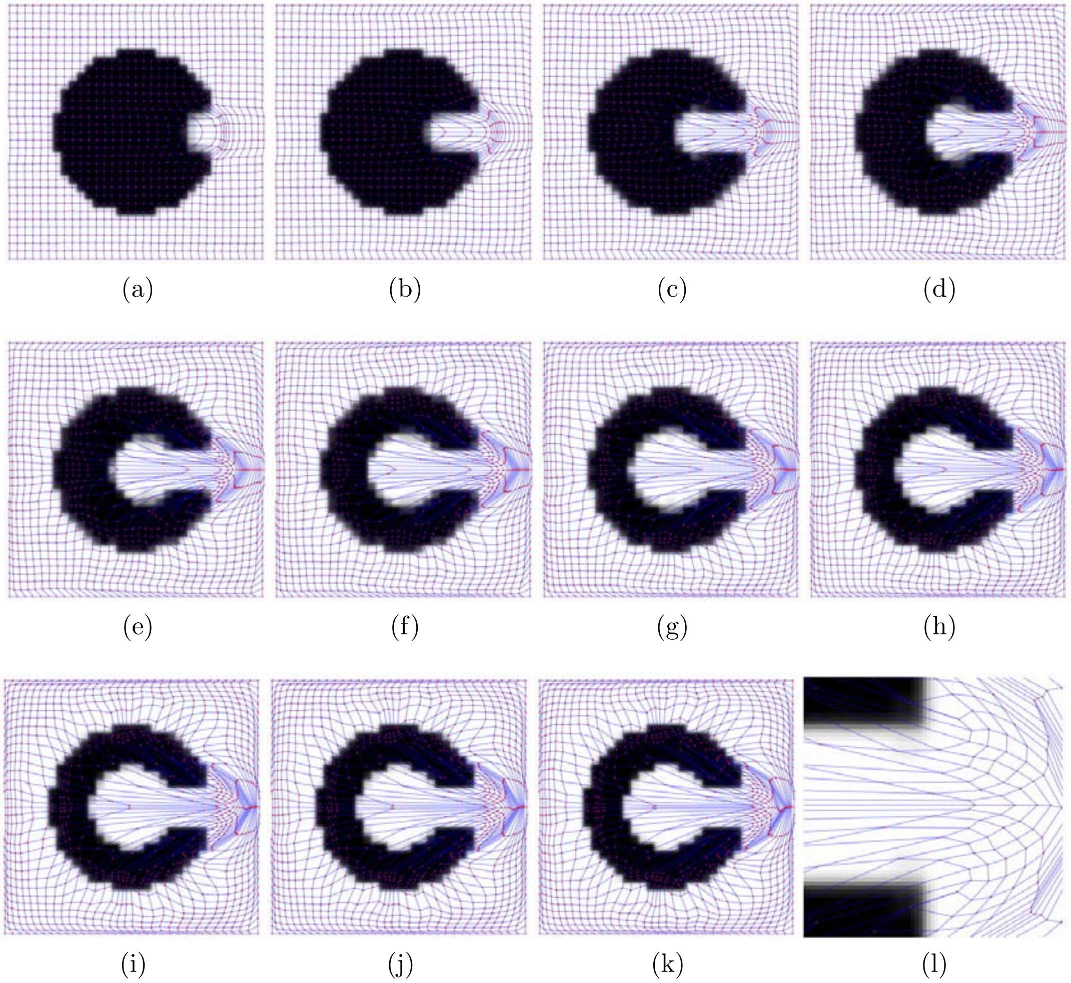
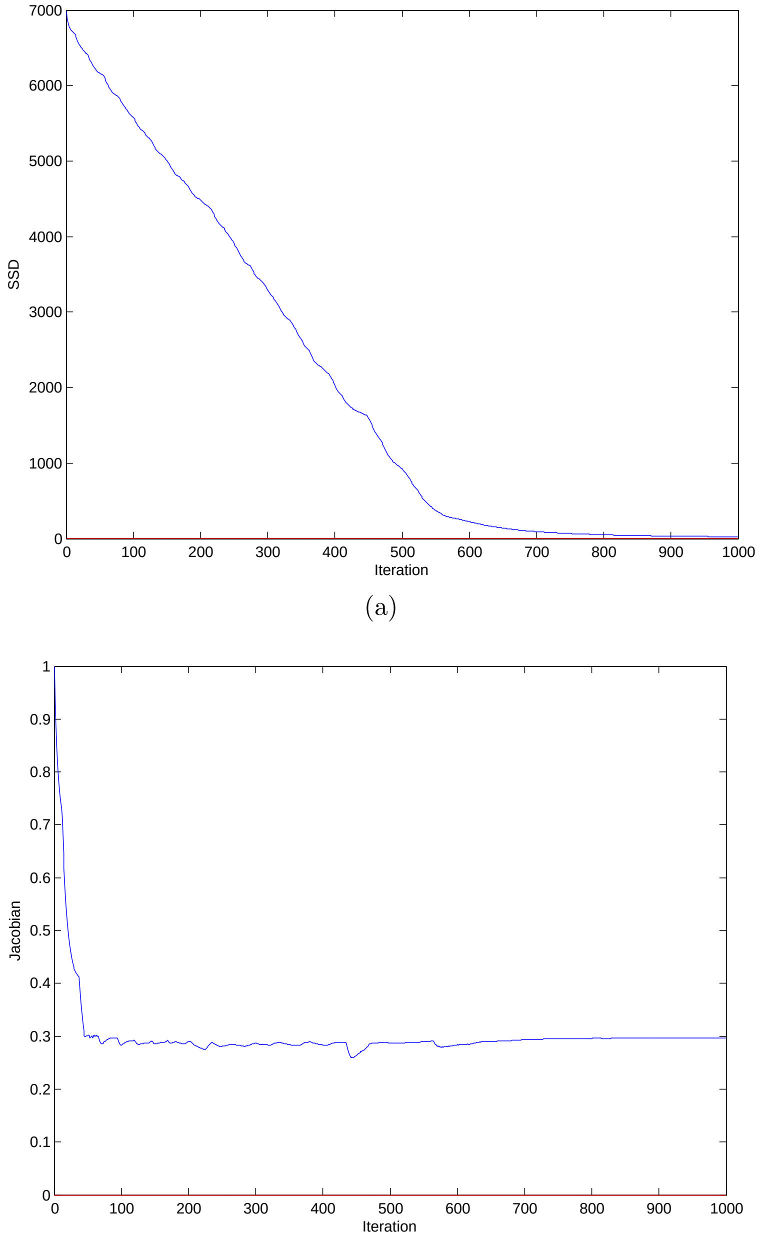


Figure 4.18. • to C results of the diffusion registration with proposed regridding method, $\alpha = 1$, $\tau = 0.01$; (a)-(k) intermediate results for k=80(80)800,1000 (l) zoom in the grid.



(a)

(b)

Figure 4.19. • to C results of the diffusion registration with proposed regridding method, $\alpha = 1$, $\tau = 0.01$; The Jacobian threshold is set to be 0.3, (a) SSD and (b) Jacobian measurement for 960 iterations (SSD=24.790118, J=0.296416 at the 960th iteration).

4.1.5 Curvature Registration: • to C

In Fig. 4.20, we show the results of using the original curvature registration algorithm. In this example, the parameters of the curvature registration are $\alpha = 10$, $\tau = 0.1$. The grid is folded at the 960th iteration as shown in Fig. 4.20k and Fig. 4.20l. Fig. 4.21a shows the improvement of SSD, however the Jacobian becomes negative (folding) at the 1037th iteration as show in Fig. 4.21b.

The results of maximum regridding frequency of the curvature registration is shown in Fig. 4.22. SSD is improved from 7000 to 20, however the grid is folded at the 732nd iteration as shown in Fig. 4.22p and Fig. 4.23b.

To show the effects of the proposed grid repairing scheme on the curvature registration, Fig. 4.24 displays the results from Algorithm 4 with $J_{min} = 0.3$, $J_{max} = \infty$, $\tau_{up} = 1$, and $\tau_{down} = 1$. The minimum of the Jacobian determinant value is successfully maintained around 0.3 (J_{min}) and with final value 0.292542.

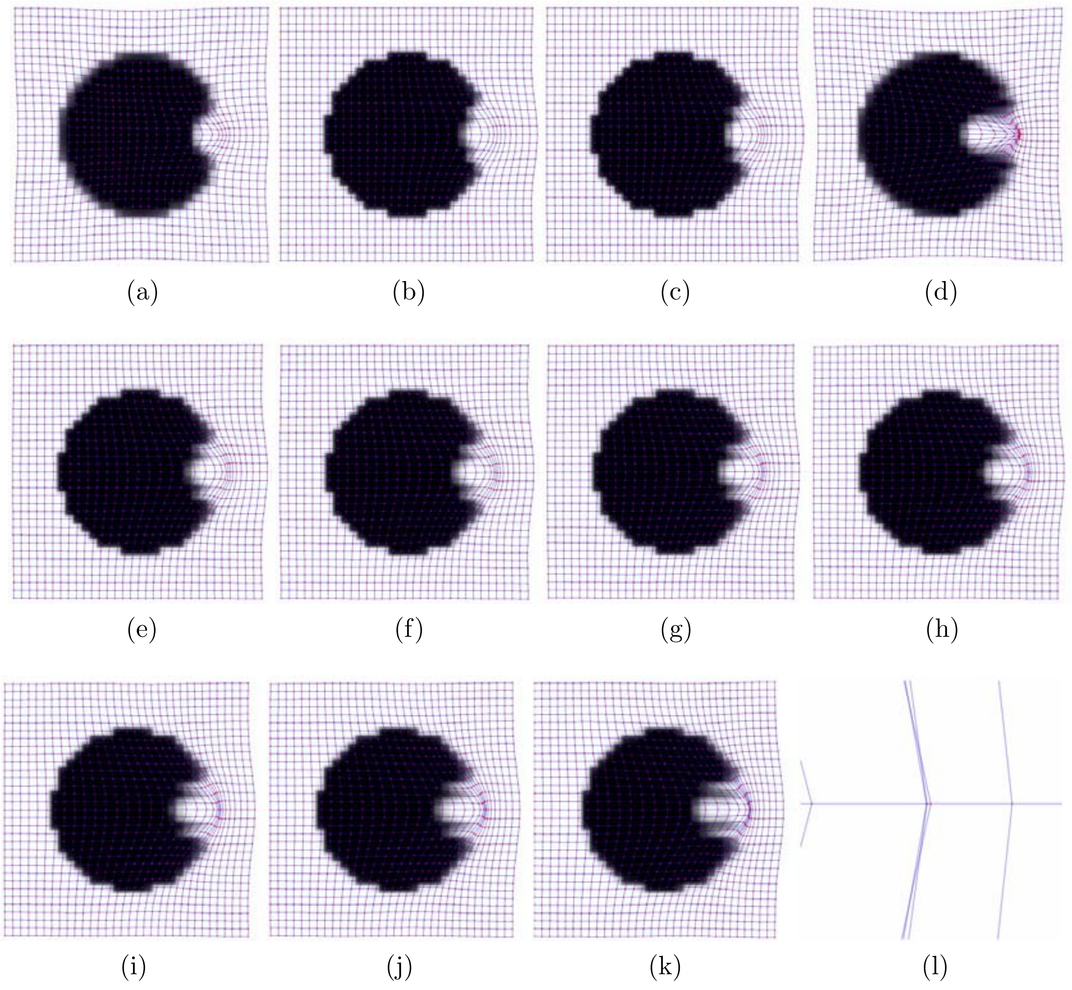
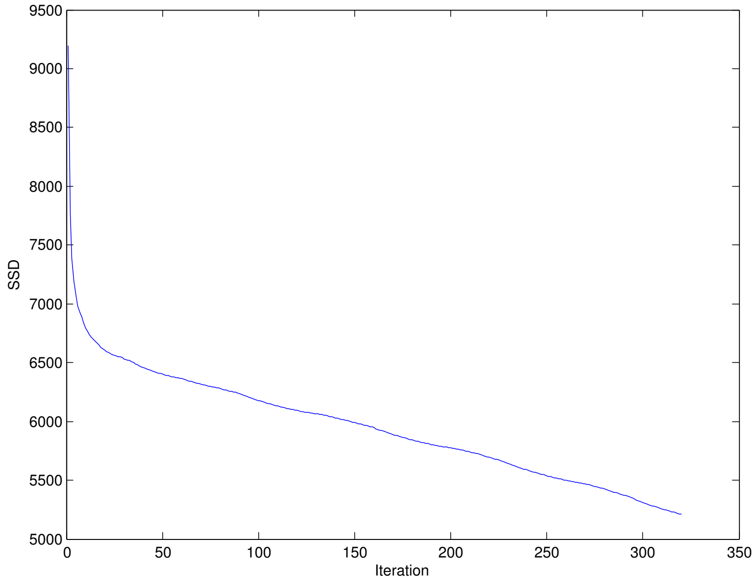
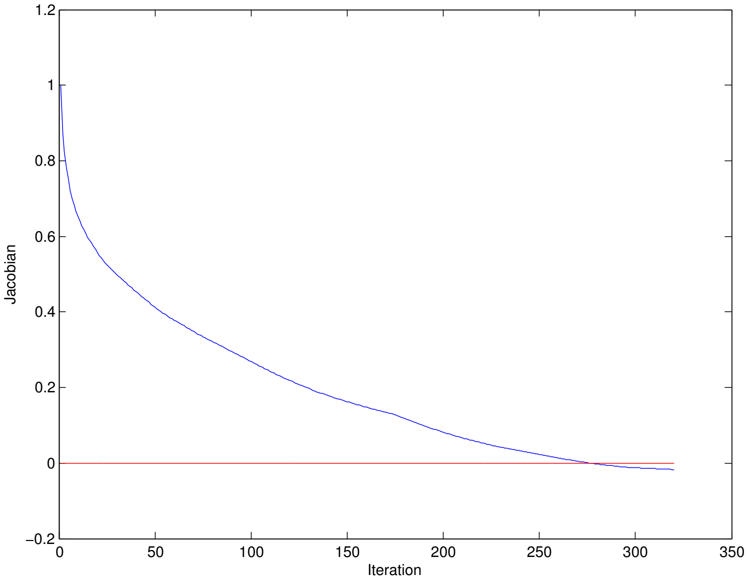


Figure 4.20. • to C results of the original curvature registration, $\tau = 0.1$, $\alpha = 10$; (a)-(k) intermediate results for k=80(80)800,1037 (l) zoom in the folded grid.



(a)



(b)

Figure 4.21. ● to C results of the original curvature registration, $\tau = 0.1$, $\alpha = 10$; (a) SSD and (b) Jacobian measurement for 1037 iterations (SSD=24.790118, J=-0.296416 at the 1037th iteration).

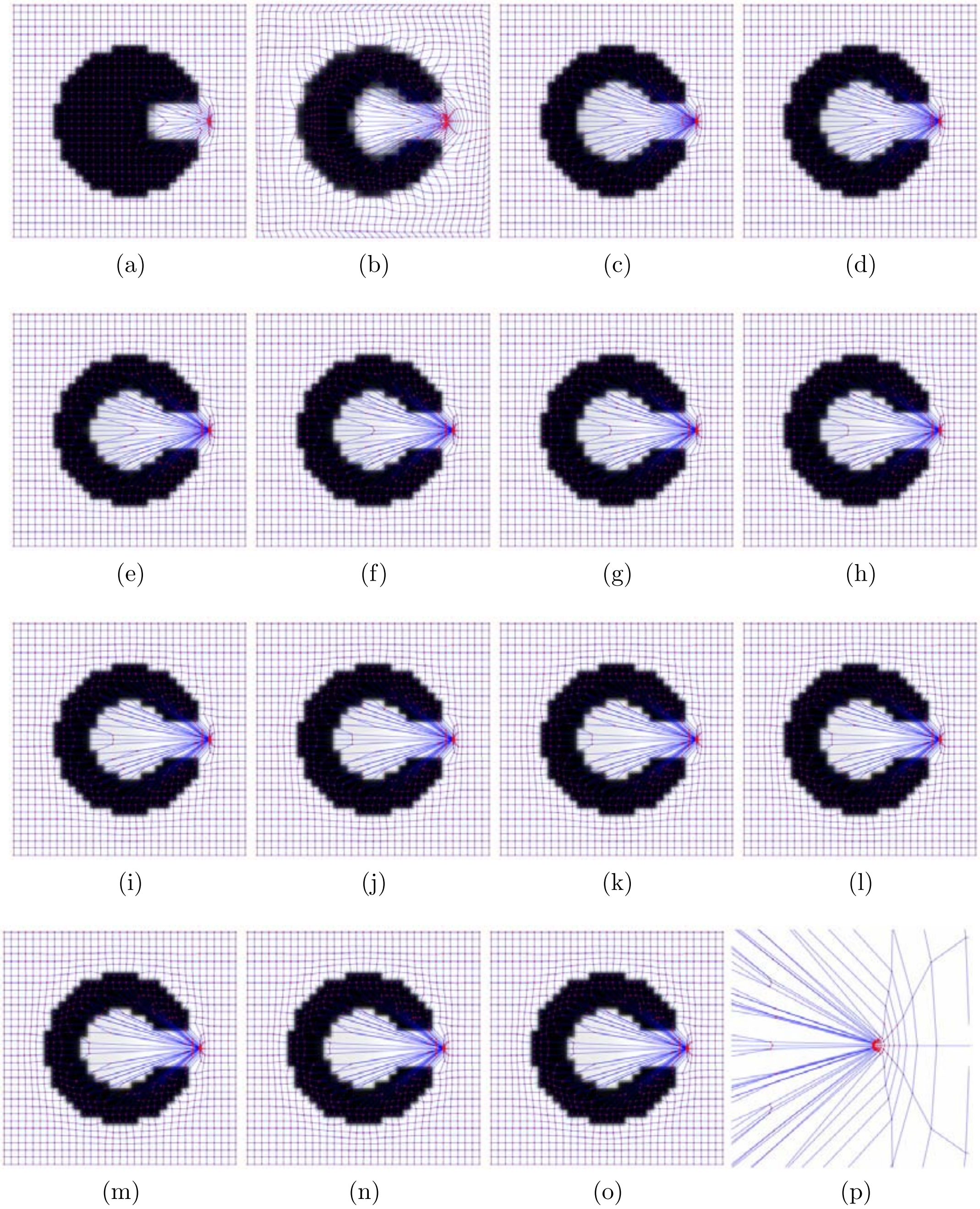
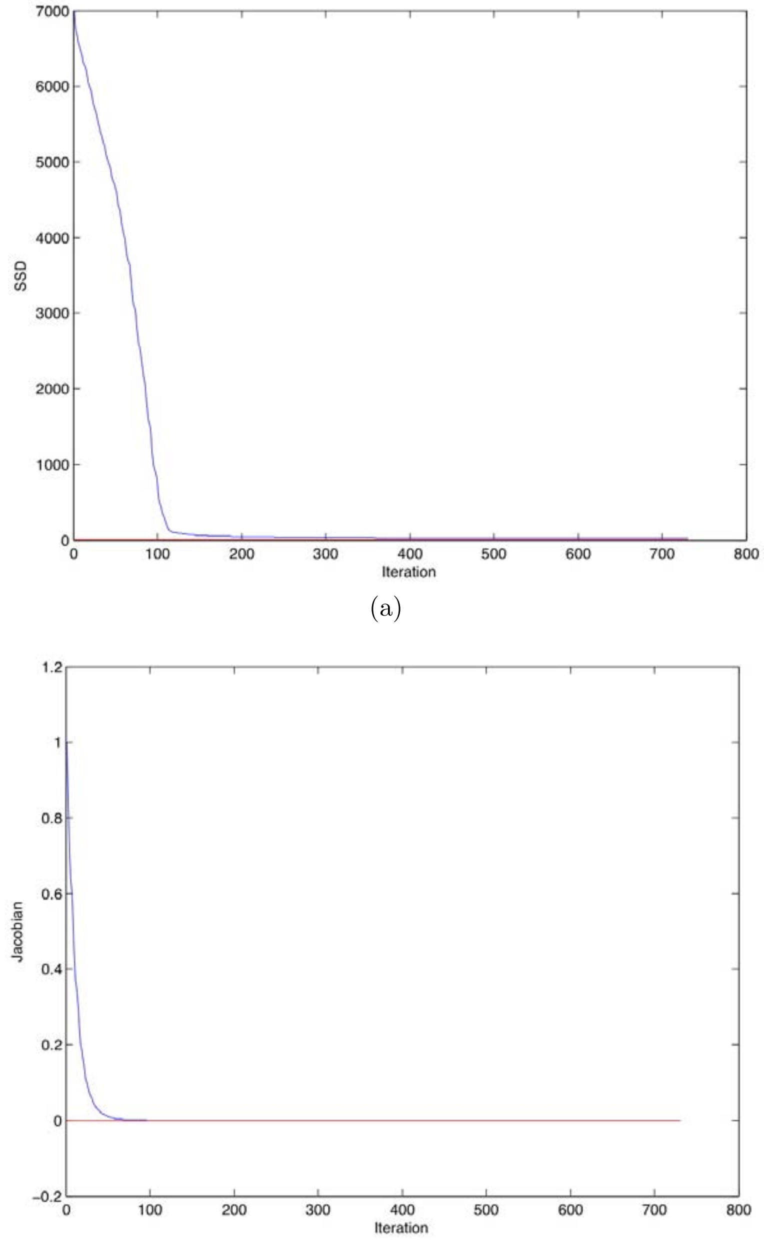


Figure 4.22. • to C results of the curvature registration with maximum regridding frequency, $\tau = 0.1$, $\alpha = 10$; (a)-(o) intermediate results for k=50(50)700,732 (p) zoom in the grid.



(a)

(b)

Figure 4.23. • to C results of the curvature registration with maximum regridding frequency, $\tau = 0.1$, $\alpha = 10$; (a)SSD and (b) Jacobian measurement for 732 iterations (SSD=20.050687 J=-0.000098 at the 732nd iteration).

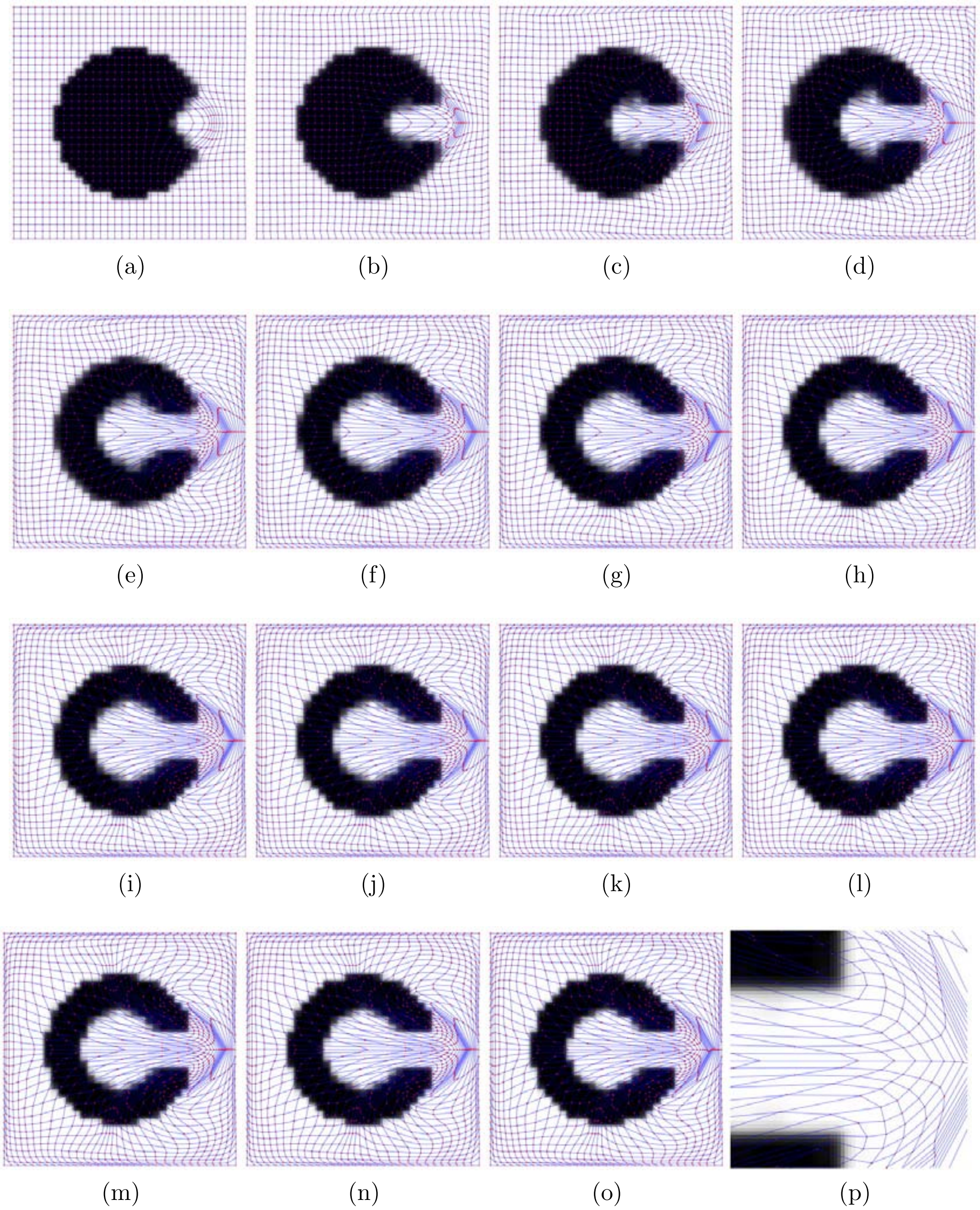
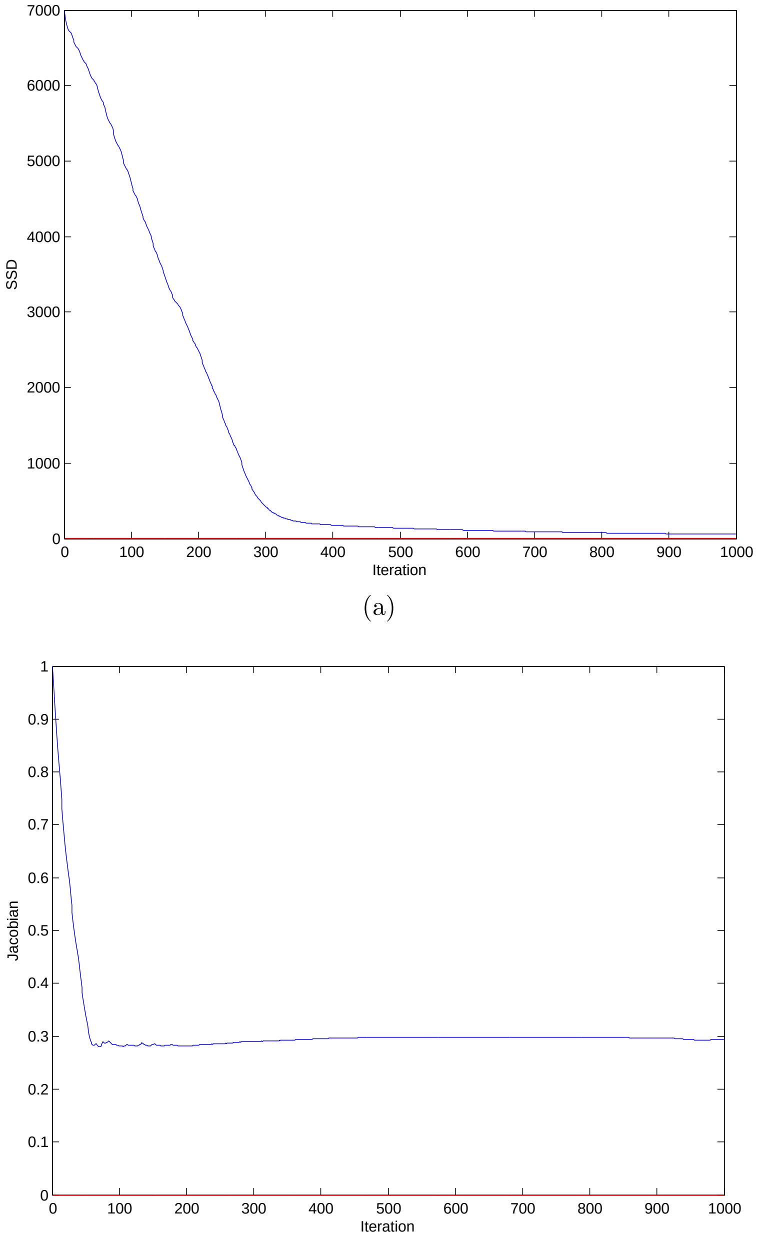


Figure 4.24. • to C results of the curvature registration with proposed regridding method, $\tau = 0.1$, $\alpha = 10$; (a)-(o) intermediate results for k=50(50)700,1000 (p) zoom in the grid.



(a)

(b)

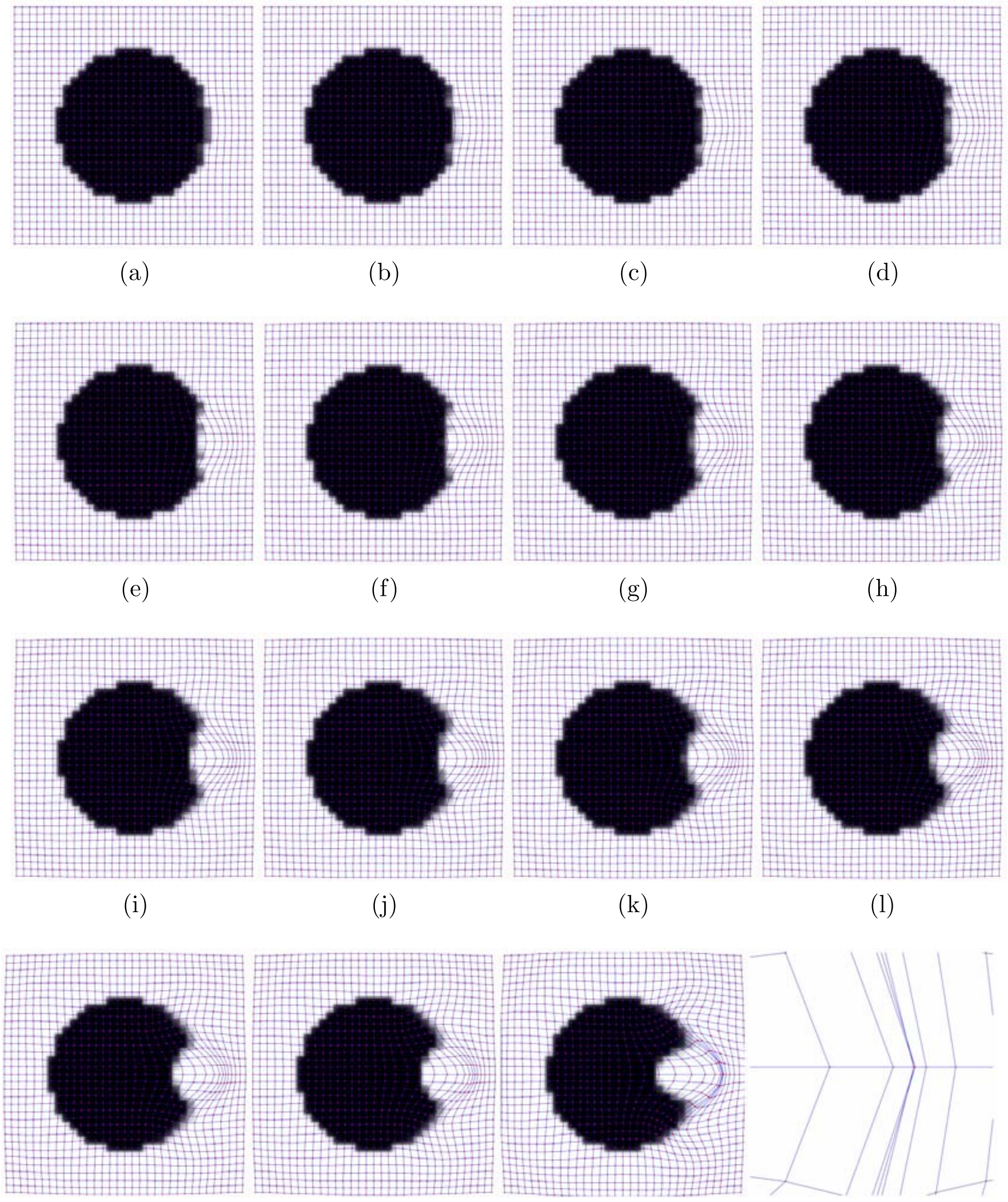
Figure 4.25. • to C results of the curvature registration with proposed regridding method, $\tau = 0.1$, $\alpha = 10$; The Jacobian threshold is set to be 0.3, (a) SSD and (b) Jacobian measurement for 1000 iterations (SSD=59.311896 J=0.292542 at the 1000th iteration).

4.1.6 Demons Registration: • to C

In Fig. 4.26, we show the results of using demons registration. In this example, the filter size of Gaussian filter is set to be 15 and the Gauss filter K_σ of characteristic width $\sigma = 3$. The grid is folded at the 664th iteration as shown in Fig. 4.26o and Fig. 4.26p. SSD is improved as shown in Fig. 4.27a, however the Jacobain becomes negative (folding) at the 664th iteration as shown in Fig. 4.27b.

The results of the maximum regridding frequency of the demons registration is shown in Fig. 4.28. SSD can be improved from 7000 to 217.462171, and the minimum of $\det(J)$ is from 1 to 0.005765. Grid is not folded as shown in Fig. 4.26p and the $\det(J)$ is positive as shown in Fig. 4.29b.

To show the effects of the proposed grid repairing scheme on the demons registration, Fig. 4.30 displays the results from Algorithm 4 with $J_{min} = 0.3$, $J_{max} = \infty$, $\tau_{up} = 1$, and $\tau_{down} = 1$. The minimum of the Jacobian determinant value is successfully maintained around 0.3 (J_{min}) and with final value 0.296351.



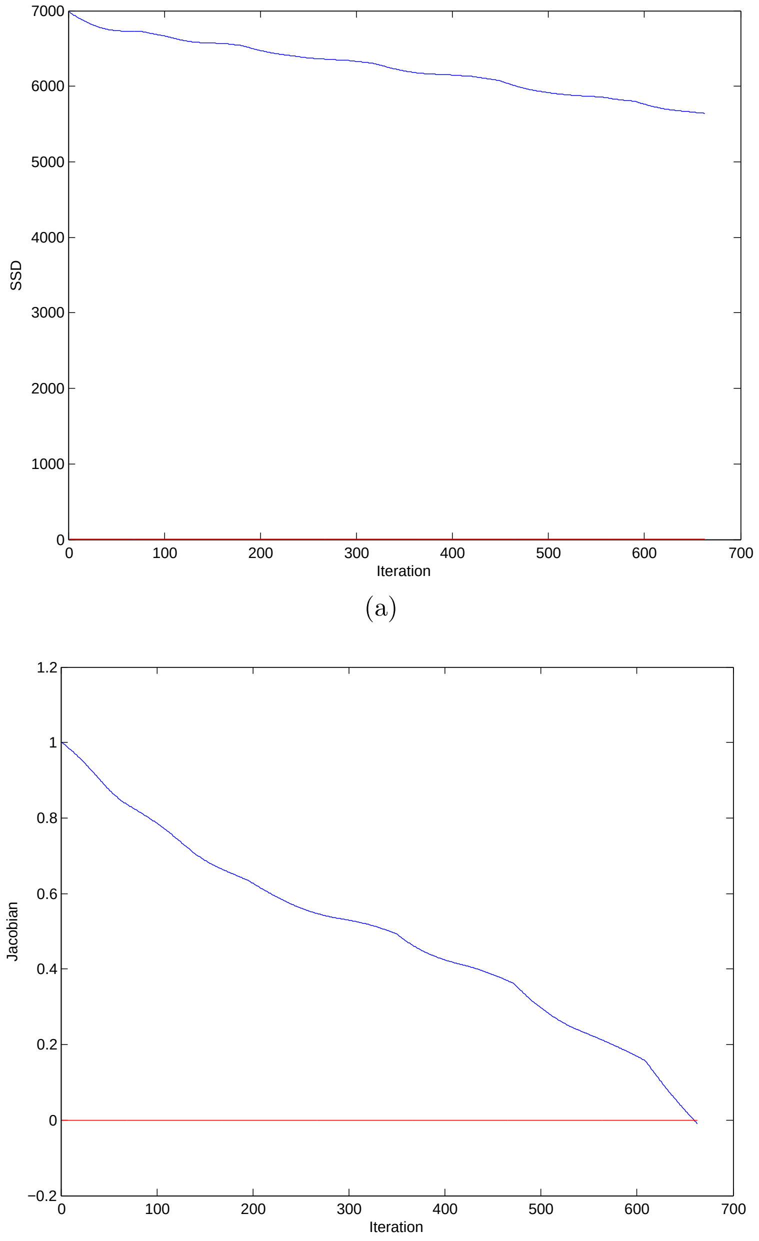
(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

Figure 4.26. • to C results of the original demons registration, $sigma = 3$, $filtersize = 15$; (a)-(o) intermediate results for k=30(30)420, 664 (p) zoom in the folded grid.



(a)

(b)

Figure 4.27. • to C results of the original demons registration, $sigma = 3$, $filtersize = 15$; (a) SSD and (b) Jacobian measurement for 664 iterations (SSD=5641.365001, J=-0.009838 at the 664th iteration).

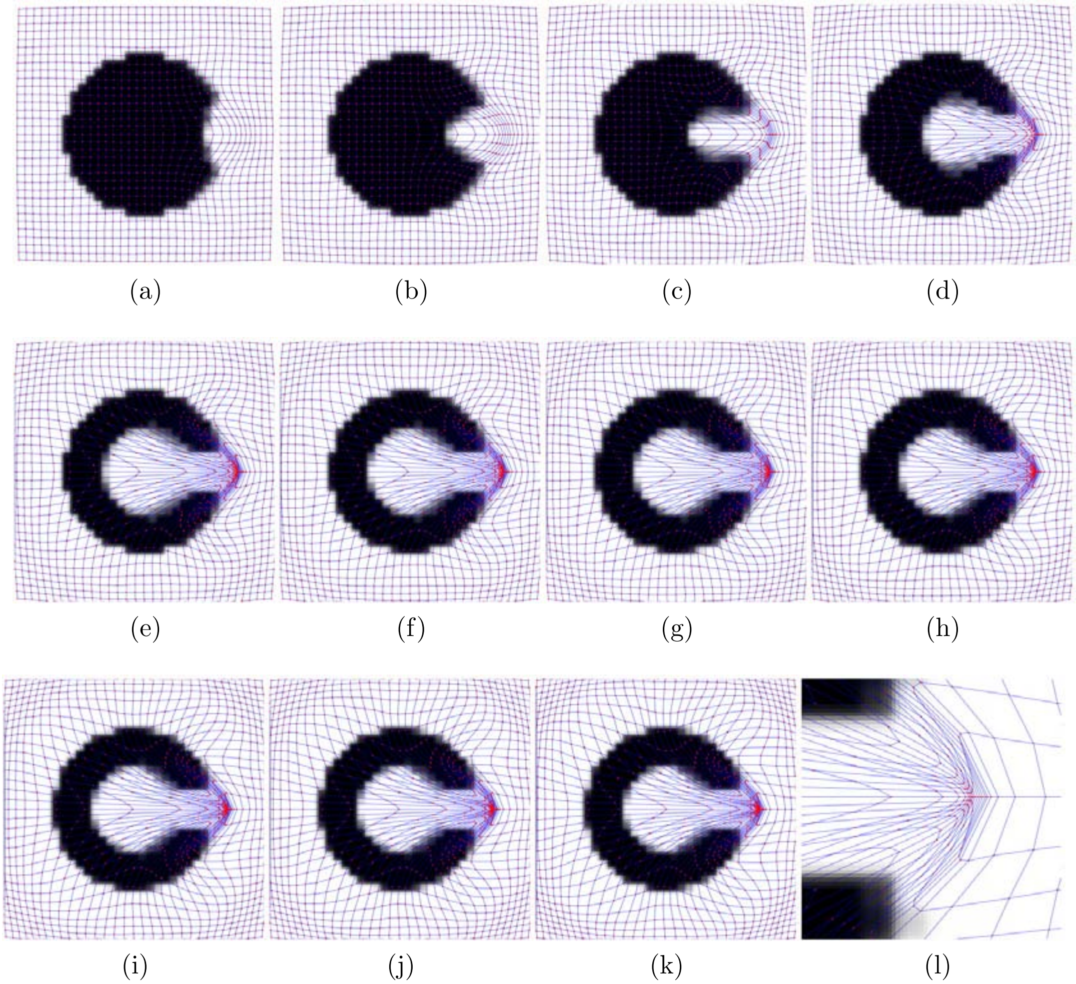
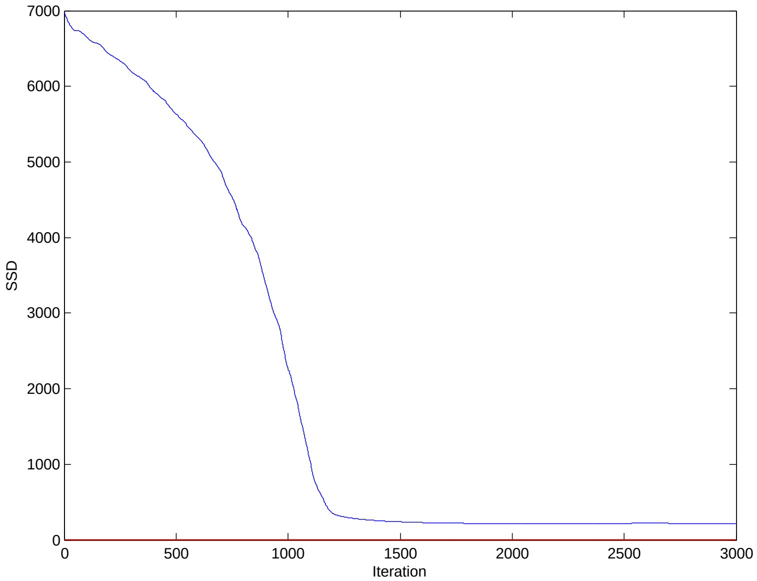
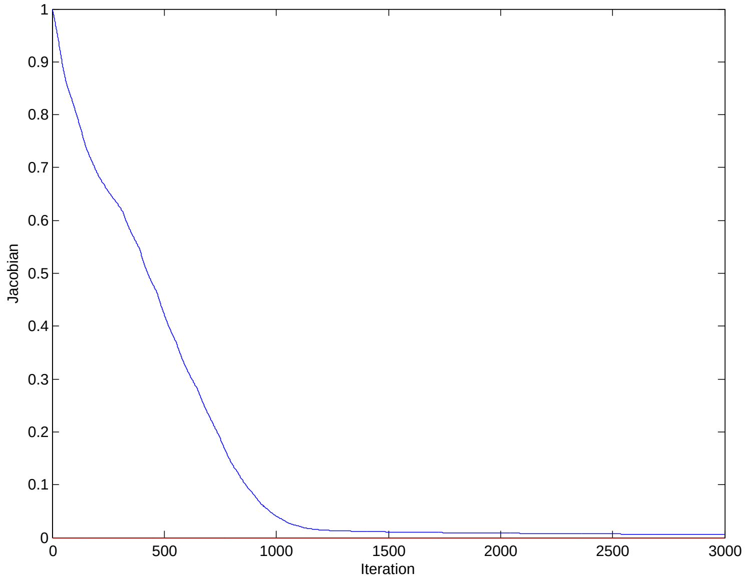


Figure 4.28. • to C results of the demons registration with maximum regridding frequency, $sigma = 3$, $filtersize = 15$; (a)-(k) intermediate results for k=250(250)2500, 3000 (l) zoom in the grid.



(a)



(b)

Figure 4.29. • to C results of the demons registration with maximum regridding frequency of, $sigma = 3$, $filtersize = 15$; (a) SSD and (b) Jacobian measurement for 3000 iterations (SSD=217.462171 J=0.005765 at the 3000th iteration).

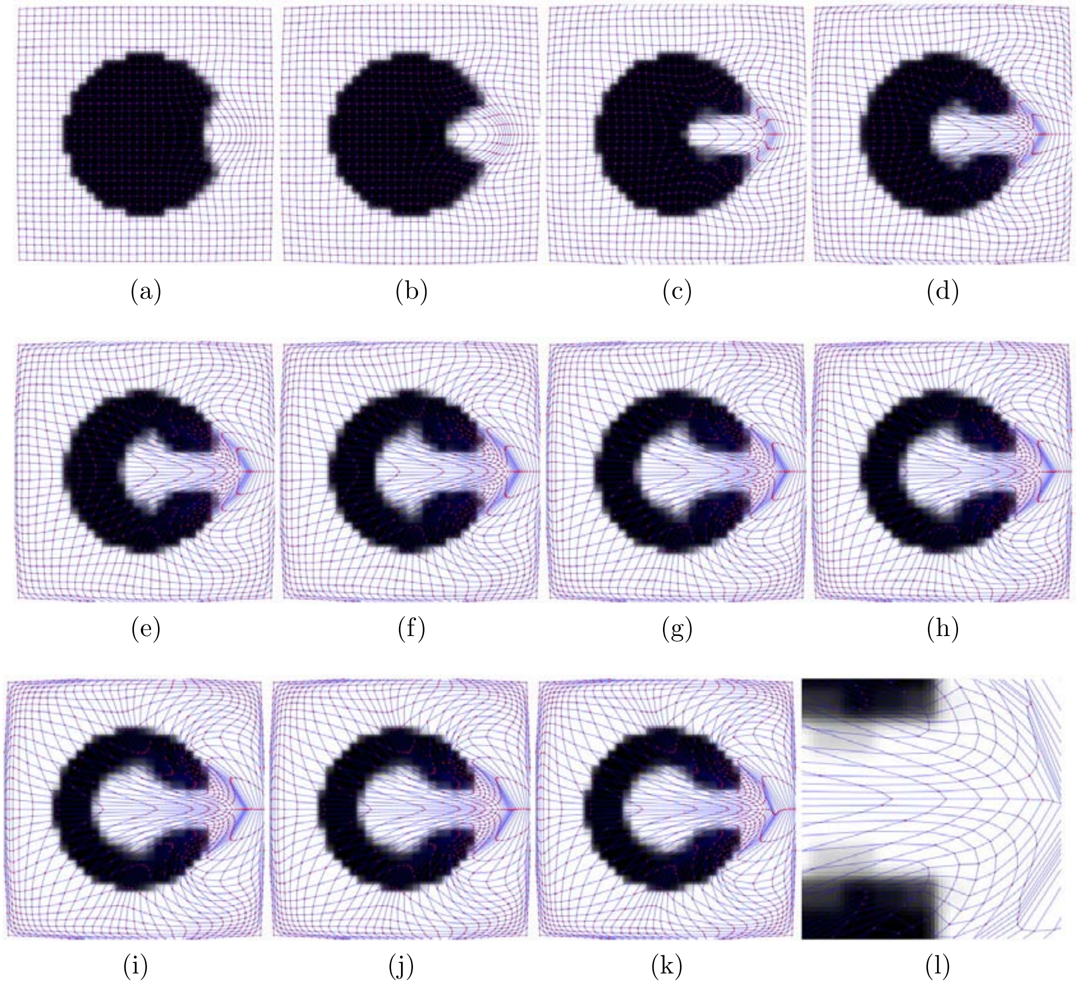
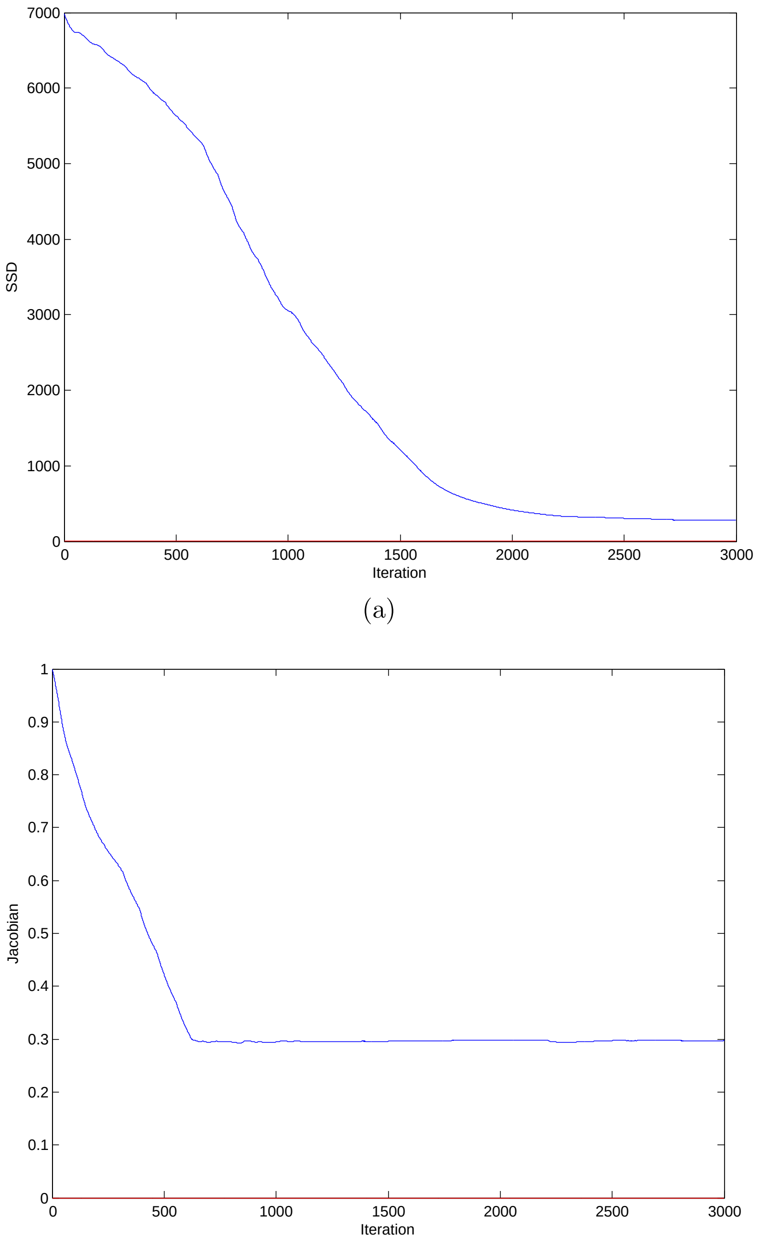


Figure 4.30. • to C results of the demons registration with proposed regridding method, $sigma = 3$, $filtersize = 15$; (a)-(k) intermediate results for k=250(250)2500, 3000 (l) zoom in the grid.



(a)

(b)

Figure 4.31. • to C results of the demons registration with proposed regridding method, $sigma = 3$, $filtersize = 15$; The Jacobian threshold is set to be 0.3, (a) SSD and (b) Jacobian measurement for 3000 iterations (SSD=286.412614 J=0.296351 at the 3000th iteration).

Table 4.1. Summary of elastic, fluid, diffusion, curvature and demons experiments on ● to C case.

	Elastic	Fluid	Diffusion	Curvature	Demons
Original Method	Folding	Folding	Folding	Folding	Folding
Maximum Frequency Regridding	● to C without folding	● to C without folding	Folding	Folding	● to C without folding
Proposed regridding method	● to C without folding	● to C without folding	● to C without folding	● to C without folding	● to C without folding

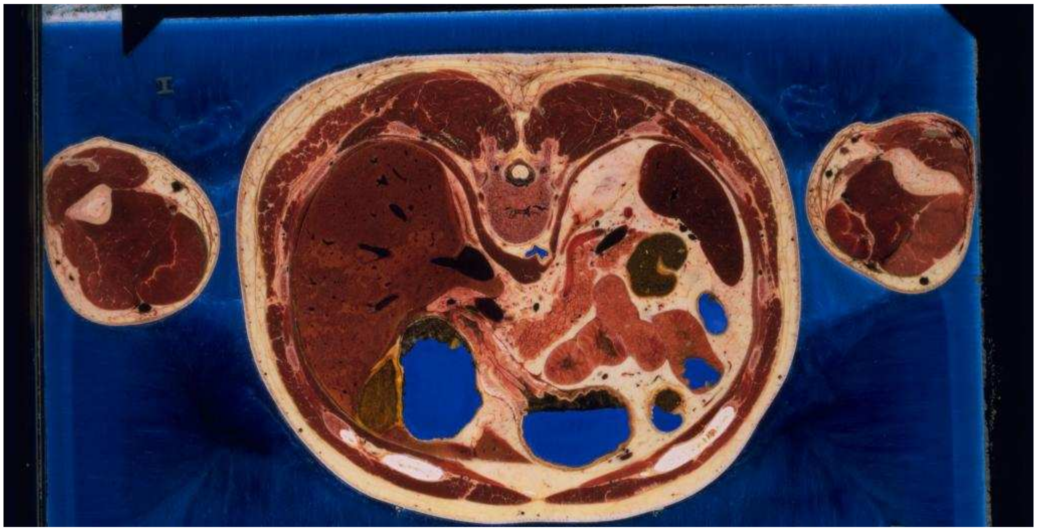


Figure 4.32. A slice of Visible Human male data.

4.2 Real Data with Ground Truth

In this section, we show experimental results using a real 2D medical image (Fig. 4.33) deformed by large synthetic deformations. The deformations are generated using thin-plate splines method with 17 by 17 control points. The original positions of the control points lie on a regular grid as shown in Fig. 4.33e. Then they are altered in a deterministic manner controlled by a parameter α ranging from 1 to 100 characterizing the degree of deformation. The parameter α is denoted as the deformation parameter thereafter in this dissertation. The bigger the α value, the larger the defeformation. The image shown in Fig. 4.33a is cropped from the red band of a slice of Visible Human male data (Fig. 4.32) in [30]. The size of the image was scaled down to 129 by 129. This image was served as the deformable template to match a sequence of reference images obtained from it with different deformation parameter values. The set of reference images were generated by warping it using the deformations with increasing deformation parameter values α from 1 to 100. Both template and reference images were added with white Gaussian

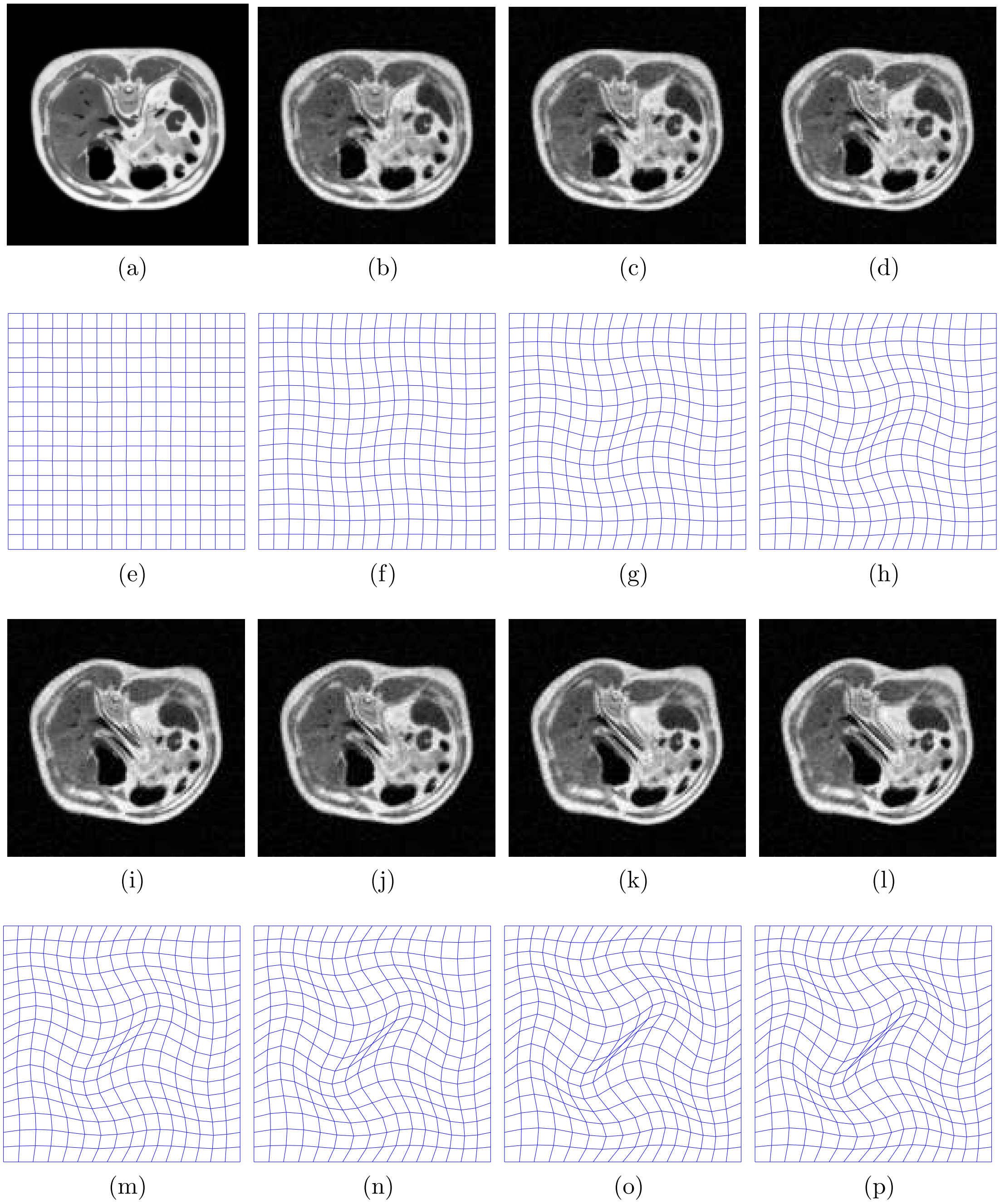


Figure 4.33. (a) A slice of Visible Human male data used as the deformable template. (b) Image warped from the deformation filed in (e). (c) Image warped from the deformation filed in (f). (d) Deformation field of initial position. (e) Final positions of deformation field by setting deformation parameter $\alpha = 1$. (f) Final positions of deformation field by setting deformation parameter $\alpha = 100$.

noise. Fig. 4.33b, 4.33c, 4.33d, 4.33i, 4.33j, 4.33k, and 4.33l show reference images deformed by α = 16, 31, 46, 61, 76, 91 and 95 respectively. Registration results recorded for further analysis include the similarity measure (SSD in our case), mean and maximum warping indices of the recovered transformations. The warping index is defined as the norm of the difference between the recovered deformation field and the ground truth [31] within the data region (homogeneous background are excluded). The definition of the mean warping index is defined as the following,

$$\bar{w} = \frac{1}{\|T\|} \sum_{\varphi \in T} \|\phi(x) - \phi^{+}(x)\| \tag{4.1}$$

And the maximum warping index is defined as the following,

$$w_{\max} = \max\left(\|\phi(x) - \phi^{+}(x)\|\right) \tag{4.2}$$

where ϕ is the deformation field obtained after the image registration, and ϕ^{+} is the ground truth of the deformation field which is generated by the thin-plate warping. x is the coordinate of the grid point and T is the template image. $\|T\|$ represents the size of image T, and $\|\phi(x) - \phi^{+}(x)\|$ is the Euclidean's distance between $\phi(x)$ and $\phi^{+}(x)$. In [31], mean warping index is an appropriate metric to assess the overall quality of the registration result if the ground truth of deformation field is available. The maximum warping index can indicate the worst difference. A substantial quality of the image registration is to consider calculating only in important pixels. It is possible to determine an *a priori* mask of significant pixels by adopting the method in [31]. Examples of masking are shown in Fig. 4.34.

Four methods are compared for each registration algorithms. "*Propagation threshold*=0" represents the original registration method. The propagation of transformation will never been activated by setting the *Propagation threshold*=0 where the threshold is used for the activation of regridding by checking the minimum of

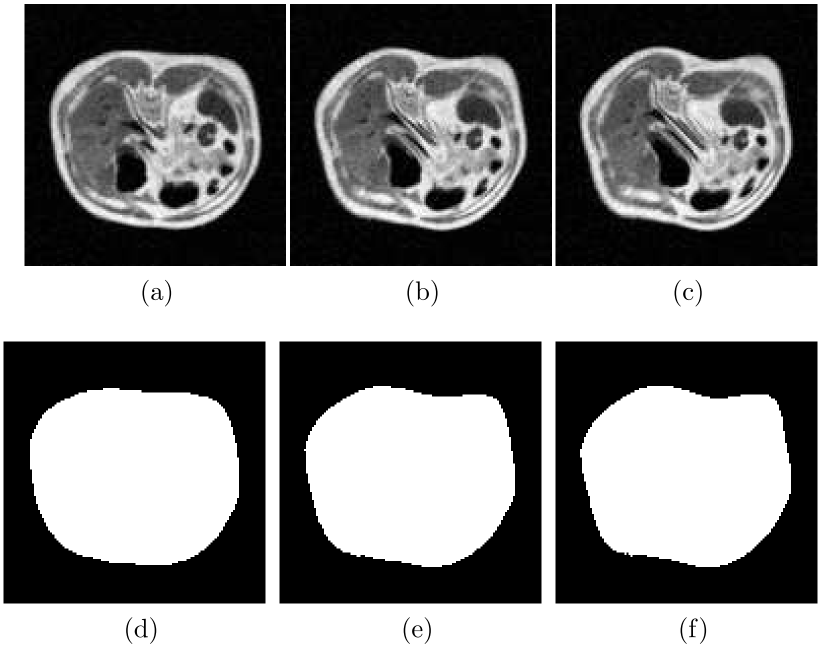


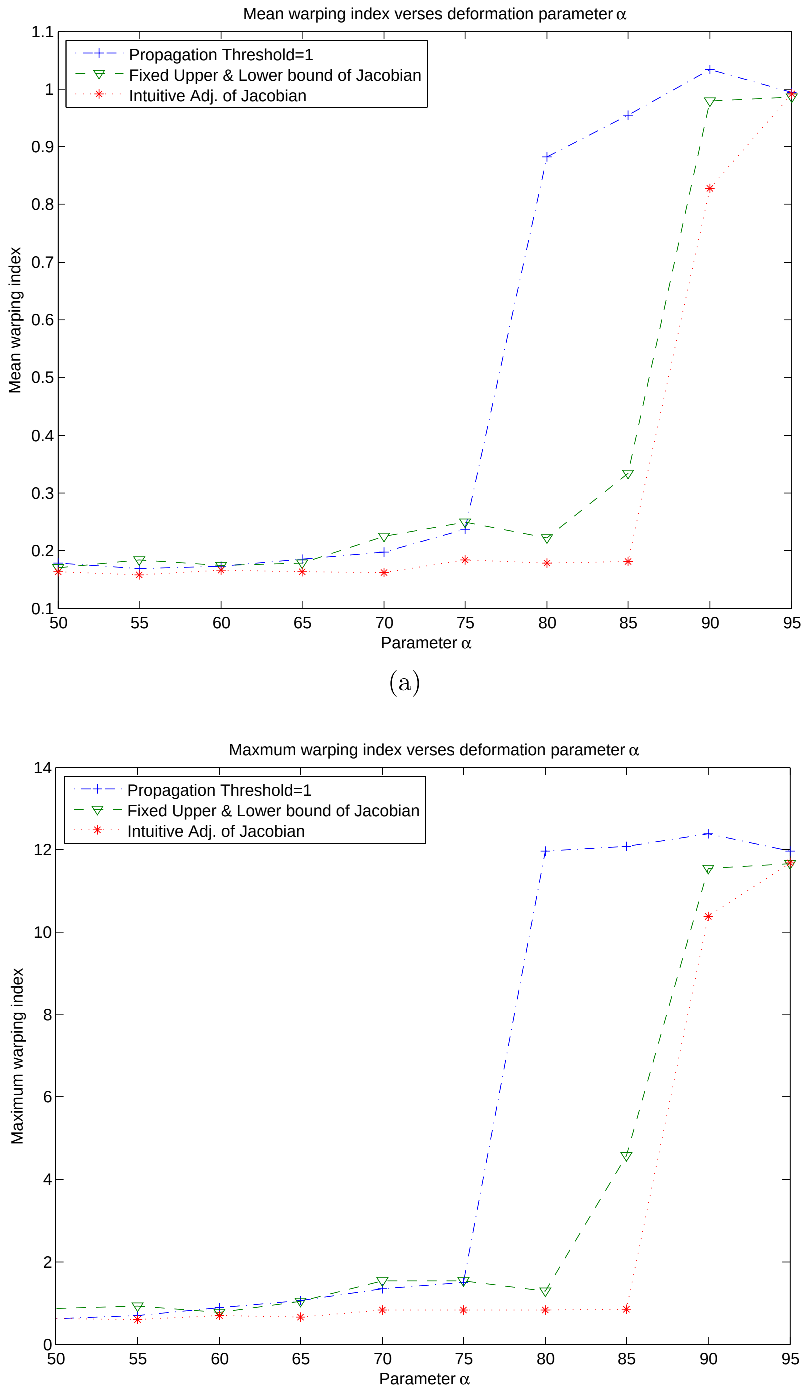
Figure 4.34. (d)-(f) are corrosponding mask of template images (a)-(c). Mean and maximum warping indices are calculated only within the mask.

Jacobian determinant. On the other hand, setting "*Propagation Threshold*=1" will lead the maximum regridding frequency. "*Fixed Upper & Lower bounds of Jacobian*" is the original method with maximum regridding frequency plus grid reparing. The "*Fixed Upper & Lower bounds of Jacobian*" is used to indicate the use of the actual maximum and minium Jacobian determinant of the transformation. However, in practice such information (actual maximum and minimum Jacobian determinant) is not available beforehand. For this reason, a method denotes as the "*Intutive Adjustment of Jacobian*" is devised. In Algorithm 4, the parameters J_{min} and J_{max} change linearly from 0.8 to 0.01 and 1.2 to 2 respectively in n stages ($n = 5$ in our case). We believe the range from 0.01 to 2 is reasonably large for most large-deformation registration applications (recall the • to C experiments). The iterations for termination criterion in each stage is $1/n$ of the value of the maximum iterations number

Algorithm 5 The Intutive Adjustment of Jacobian for 1000 iterations in 5 stages

```
1: 1st stage: 200 iterations $J_{min} = 0.8$, $J_{max} = 1.2$
2: 2nd stage: 200 iterations $J_{min} = 0.6$, $J_{max} = 1.4$
3: 3rd stage: 200 iterations $J_{min} = 0.4$, $J_{max} = 1.6$
4: 4th stage: 200 iterations $J_{min} = 0.2$, $J_{max} = 1.8$
5: 5th stage: 200 iterations $J_{min} = 0.01$, $J_{max} = 2$
```

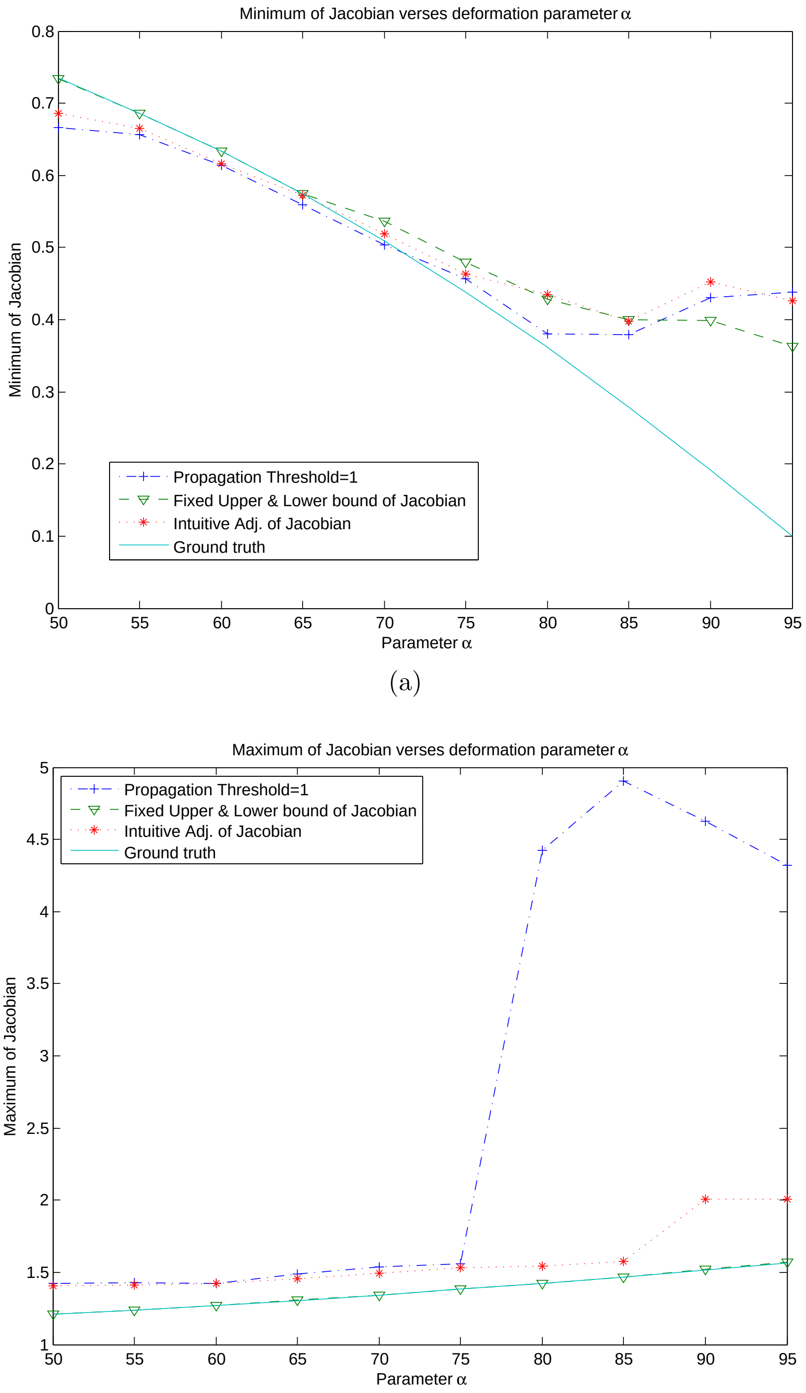
($n = 5$ and maximum iterations = 1000 in our cases). An example is shown in Algorithm 5. Each stage of registration is terminated when either the step size (τ in Algorithm 4) drops below the pre-defined value 0.01 or the maximum number of iterations (200) is reached and registration continues in the next stage with a wider range for the Jacobian determinant until the last stage is completed. Eq. (2.11) is used for transformation concatenation in our experiments.



(a)

(b)

Figure 4.35. Experimental results by using elastic algorithm. $\mu = 800$, $\lambda = 100$ (a)Mean warping index (b)Maximum warping index.



(a)

(b)

Figure 4.36. Experimental results by using elastic algorithm. $\mu = 800$, $\lambda = 100$ (a)Minimum of the Jacobian determinant (b)Maximum of the Jacobian determinant.

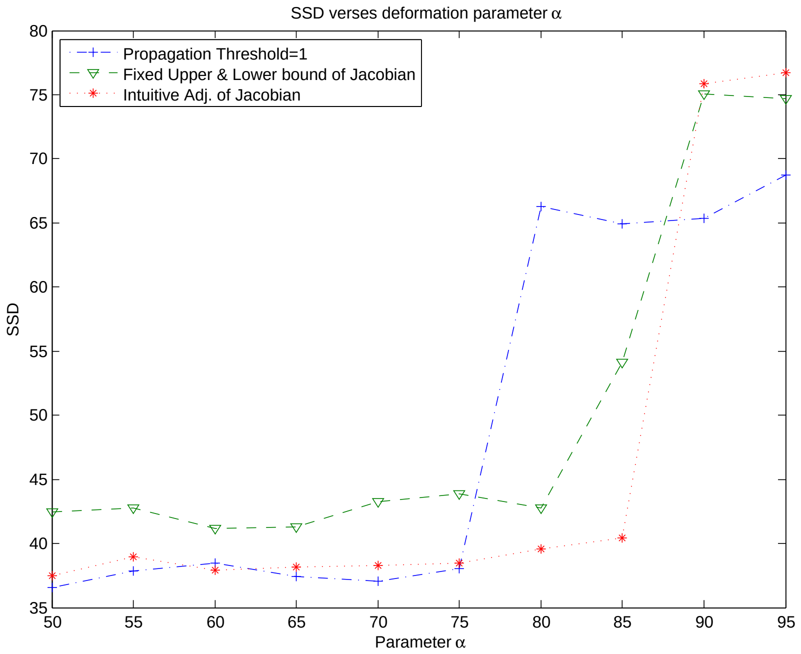
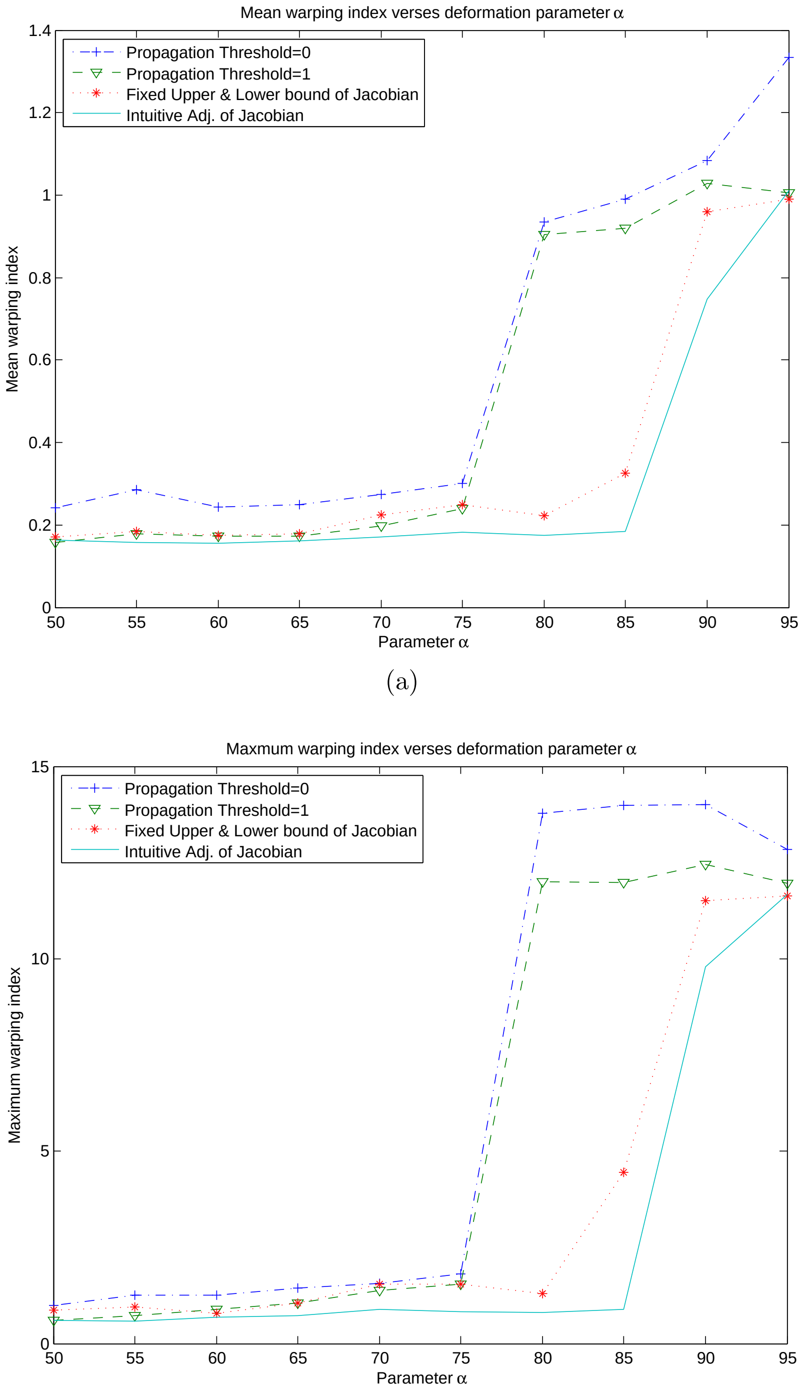


Figure 4.37. SSD measurement of elastic algorithm. $\mu = 800$, $\lambda = 100$.

Table 4.2. Summary of elastic experiments on synthetic data with ground truth

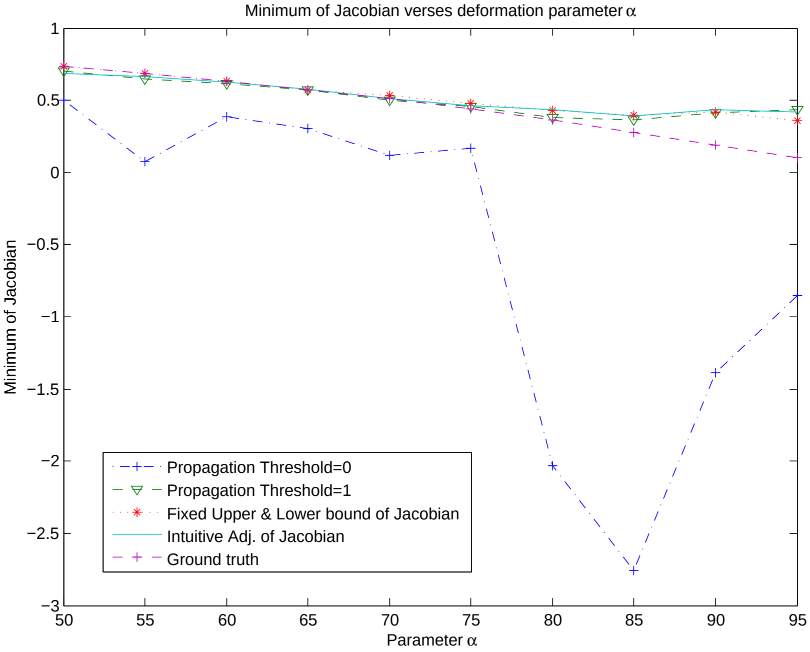
Propagation Threshold:	1	Fixed Upper and Lower Bound of Jacobian	Intuitive Adjustment of Jacobian
Max. deformation paprameter α	75	80	85
Average SSD	38.24234	42.54616	38.77102
Mean Warping Index	0.18698	0.20095	0.16857
Max. Warping Index	1.03138	1.14961	0.75396
Average number of iteration	625.42857	258.5	999.3750



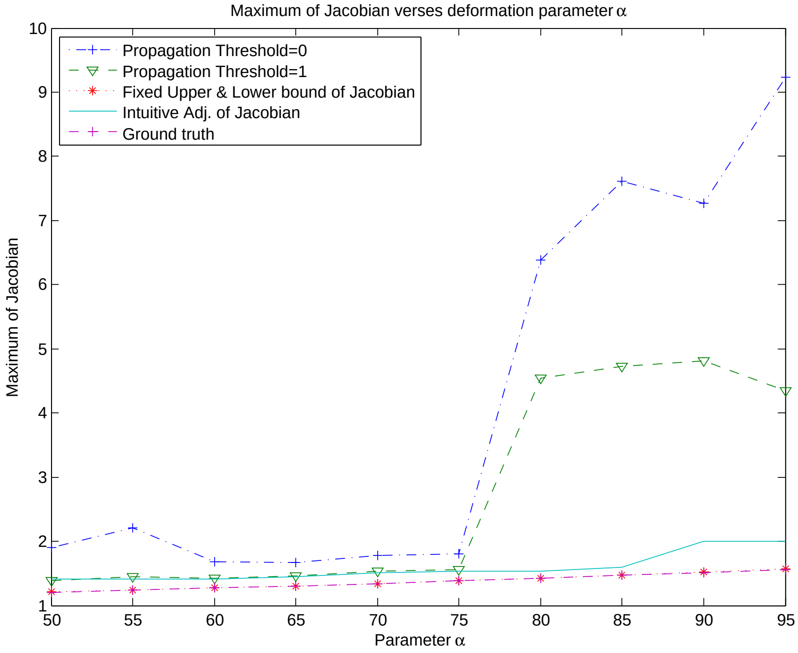
(a)

(b)

Figure 4.38. Experimental results by using fluid algorithm. $\mu = 800$, $\lambda = 100$ (a)Mean warping index (b)Maximum warping index.



(a)



(b)

Figure 4.39. Experimental results by using fluid algorithm. $\mu = 800$, $\lambda = 100$ (a)Maximum of the Jacobian determinant (b)Minimum of the Jacobian determinant.

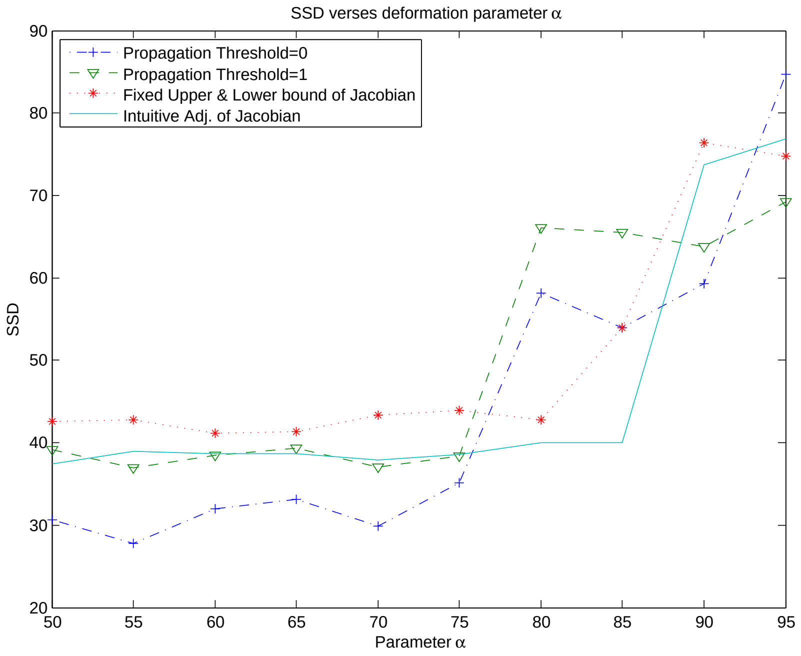
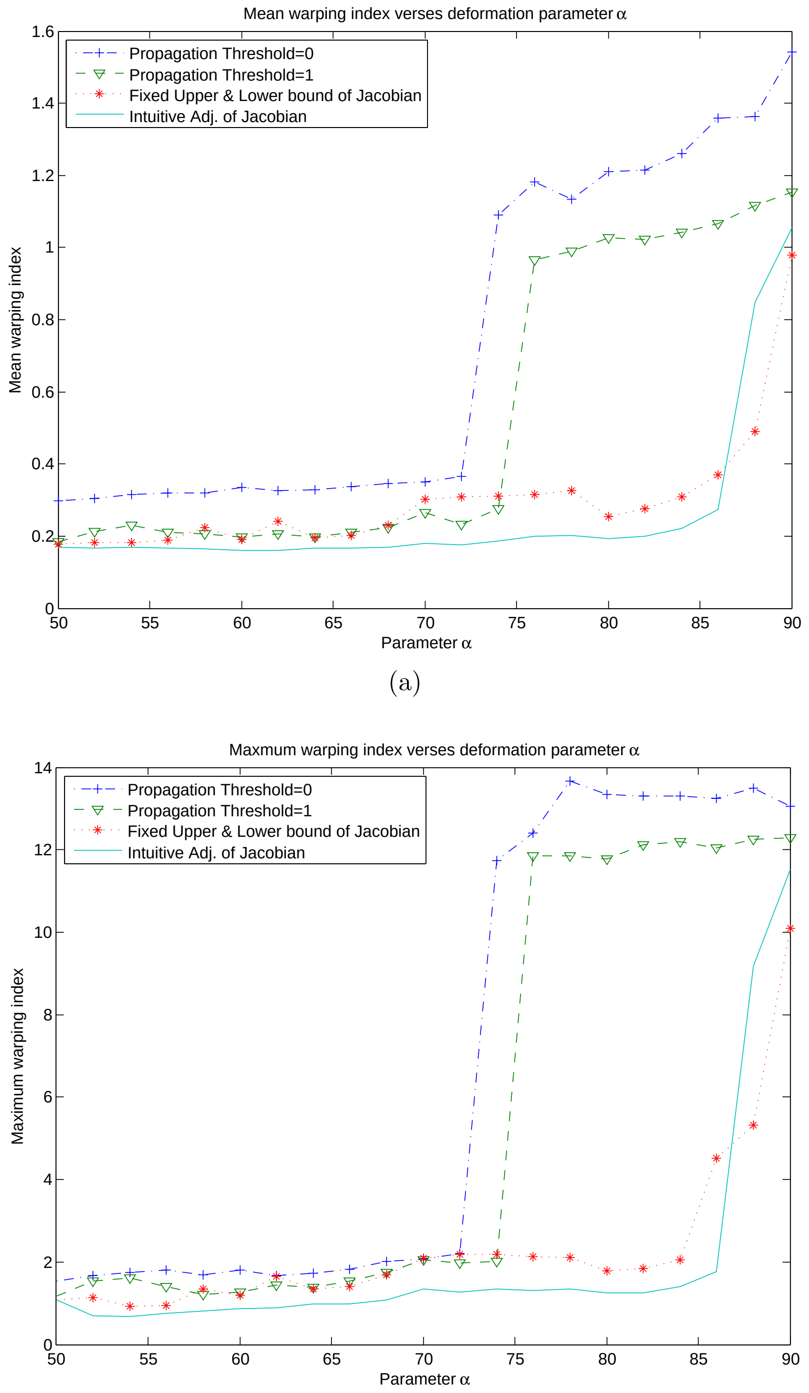


Figure 4.40. SSD measurement of fluid algorithm. $\mu = 800$, $\lambda = 100$.

Table 4.3. Summary of fluid experiments on synthetic data with ground truth

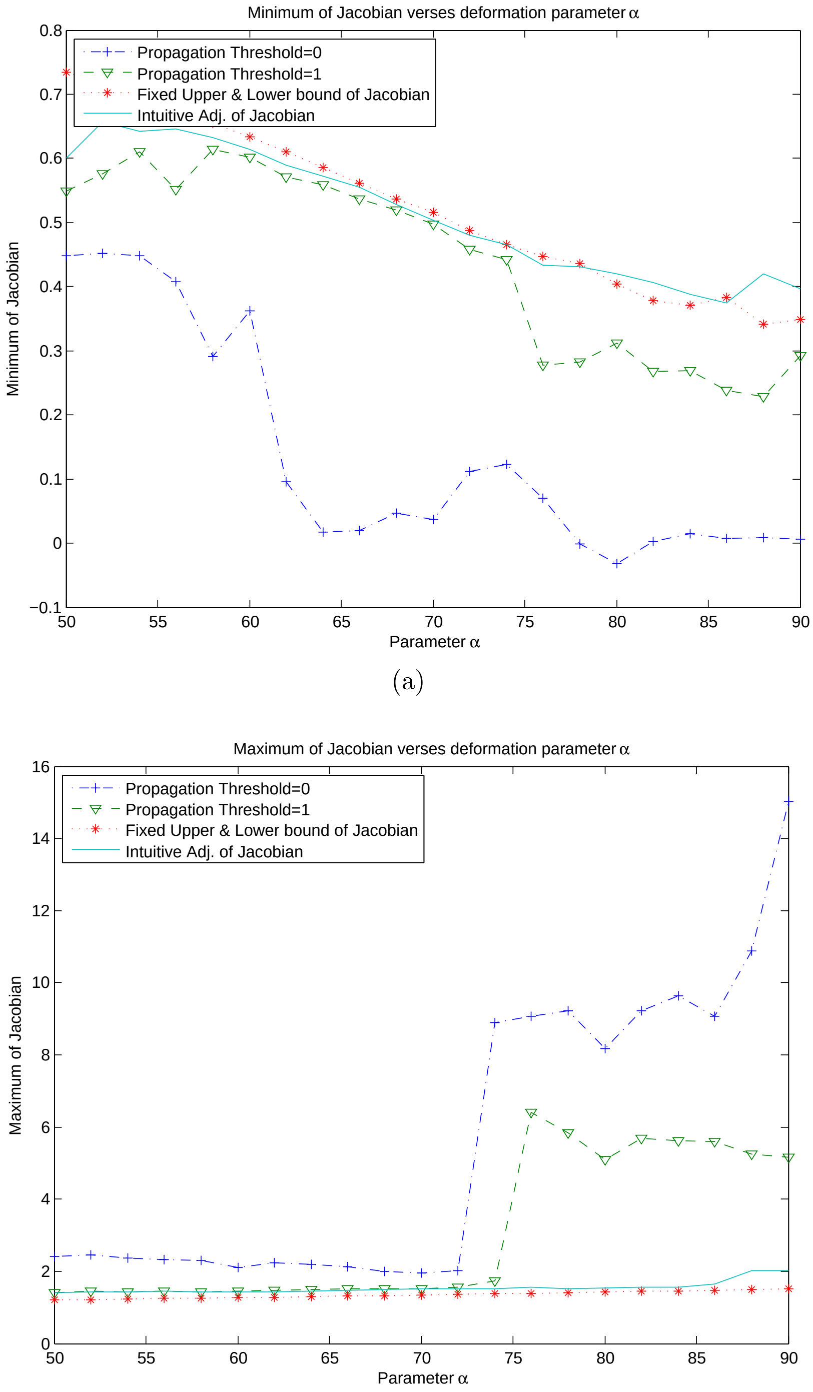
Propagation Threshold:	0	1	Fixed Upper and Lower Bound of Jacobian	Intuitive Adjustment of Jacobian
Max. deformation paprameter α	75	75	80	85
Average SSD	31.42251	38.24234	42.54616	38.77102
Mean Warping Index	0.26586	0.18698	0.20095	0.16857
Max. Warping Index	1.38759	1.03138	1.14961	0.75396
Average number of iteration	546.1667	625.42857	258.5	999.3750



(a)

(b)

Figure 4.41. Experimental results by using diffusion algorithm. $\tau = 0.02$, $\alpha = 1e3$ (a)Mean warping index (b)Maximum warping index.



(a)

(b)

Figure 4.42. Experimental results by using diffusion algorithm. $\tau = 0.02$, $\alpha = 1e3$ (a)Minimum of the Jacobian determinant (b)Maximum of the Jacobian determinant.

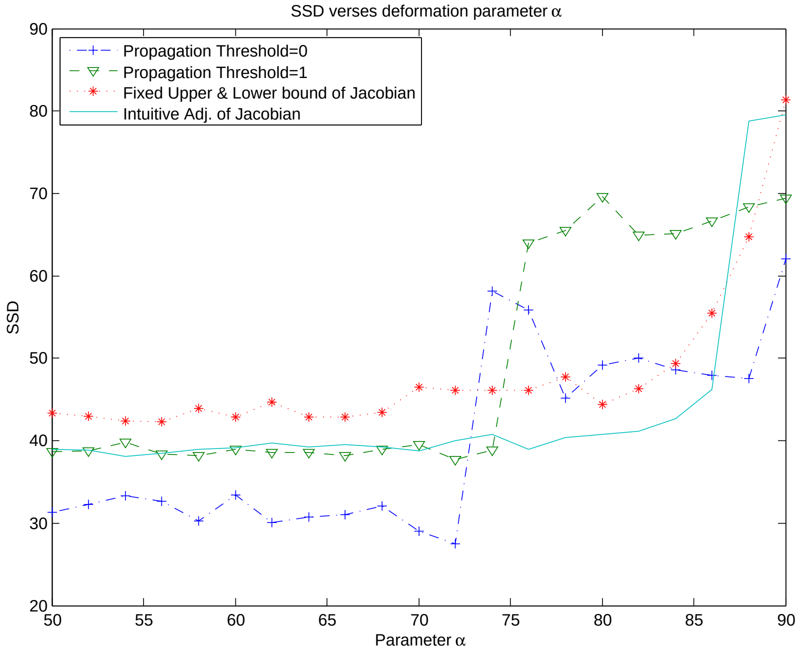
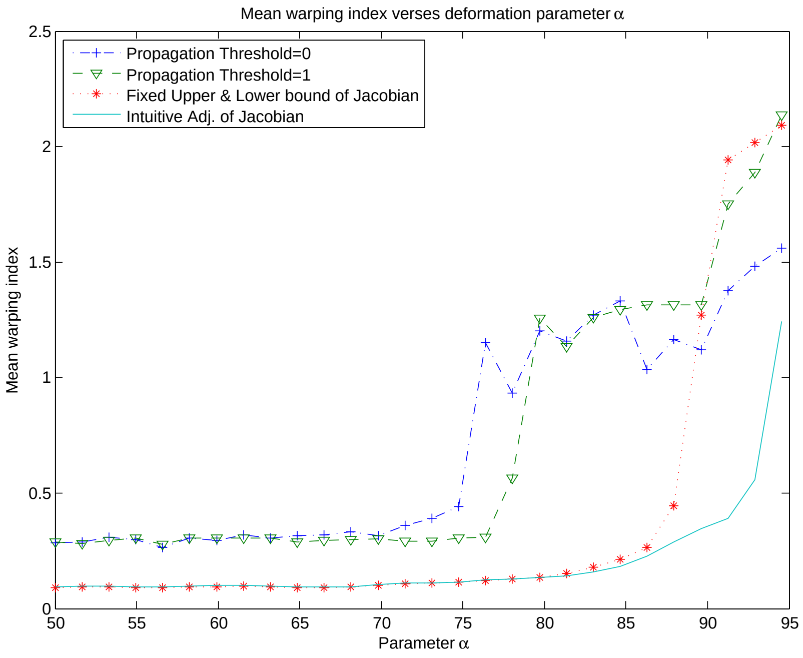


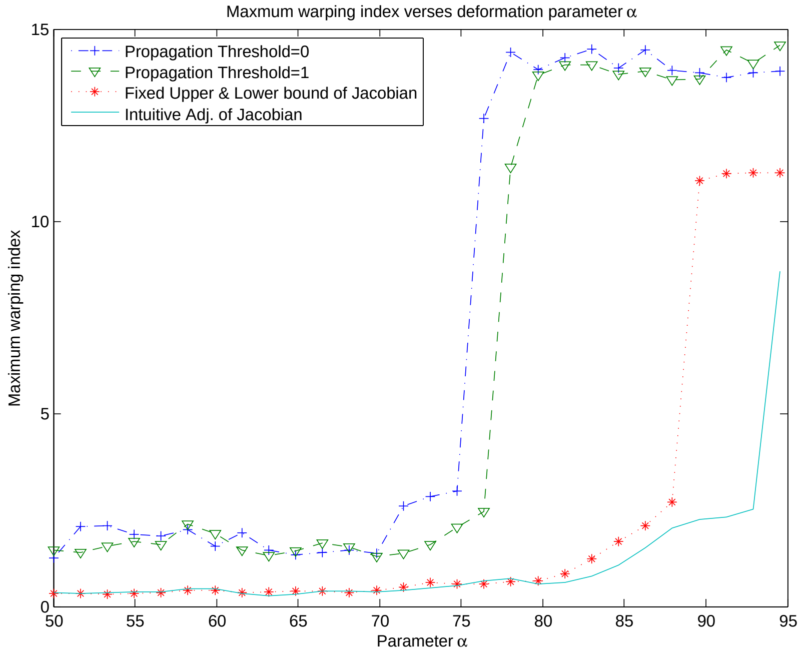
Figure 4.43. SSD measurement of diffusion algorithm. $\tau = 0.02$, $\alpha = 1e3$.

Table 4.4. Summary of diffusion experiments on synthetic data with ground truth

Propagation Threshold:	0	1	Fixed Upper and Lower Bound of Jacobian	Intuitive Adjustment of Jacobian
Max. deformation paprameter α	72	74	84	86
Average SSD	31.14391	34.45255	44.68650	40.00306
Mean Warping Index	0.32889	0.2967	0.24555	0.18357
Max. Warping Index	1.81472	1.64602	1.61646	1.11264
Average number of iteration	714.16667	805.33333	824.8	995.84211

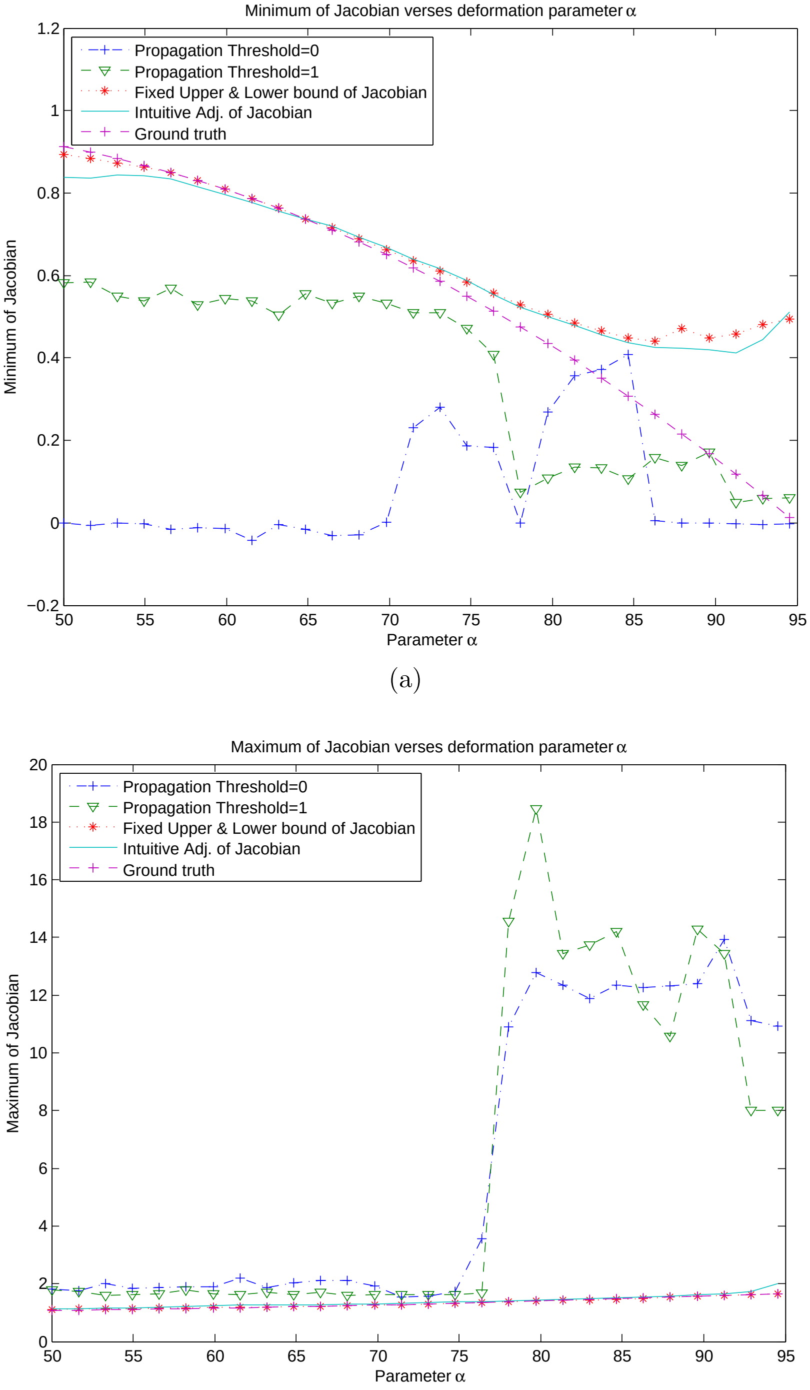


(a)



(b)

Figure 4.44. Experimental results by using curvature algorithm. $\tau = 0.5$, $\alpha = 1e3$ (a)Mean warping index (b)Maximum warping index.



(a)

(b)

Figure 4.45. Experimental results by using curvature algorithm. $\tau = 0.5$, $\alpha = 1e3$ (a)Minimum of the Jacobian determinant (b)Maximum of the Jacobian determinant.

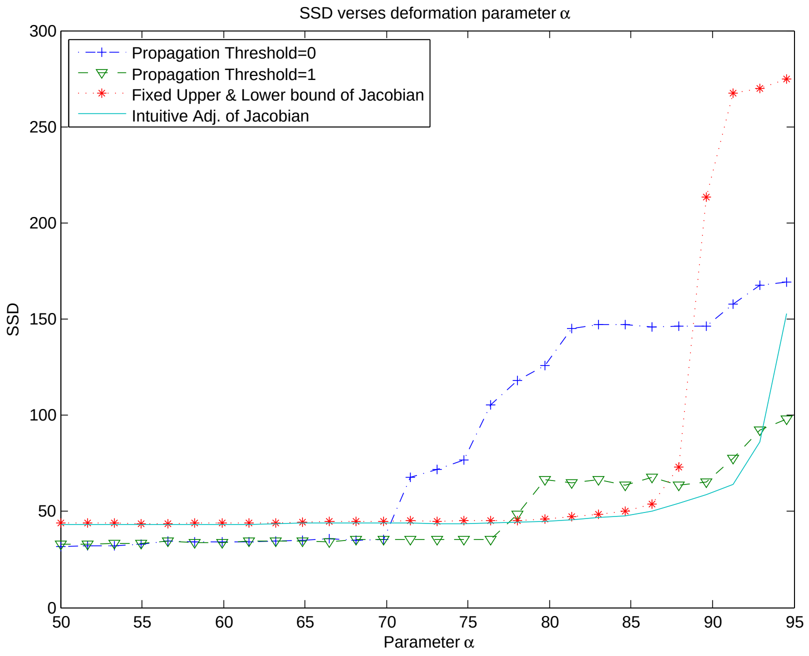
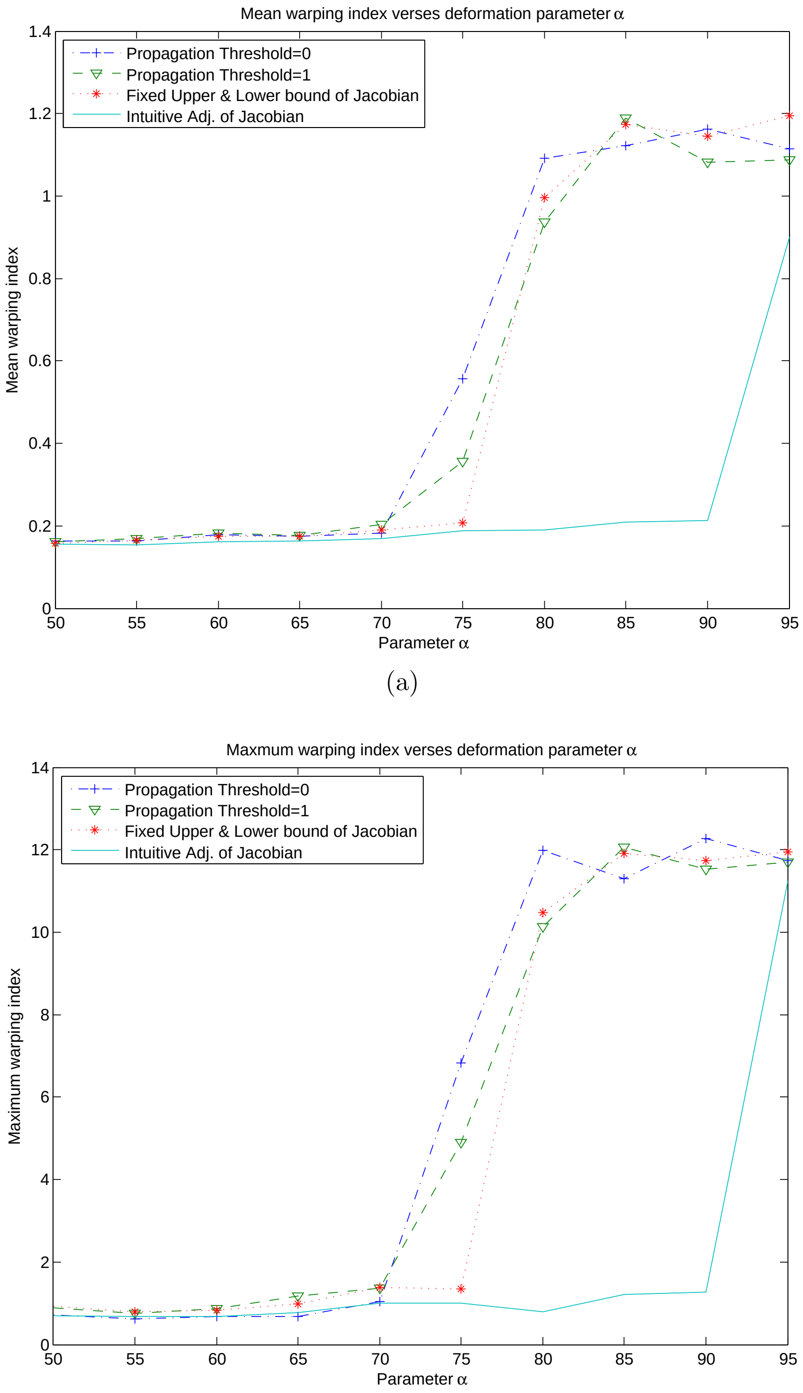


Figure 4.46. SSD measurement of curvature algorithm. $\tau = 0.5$, $\alpha = 1e3$.

Table 4.5. Summary of curvature experiments on synthetic data with ground truth

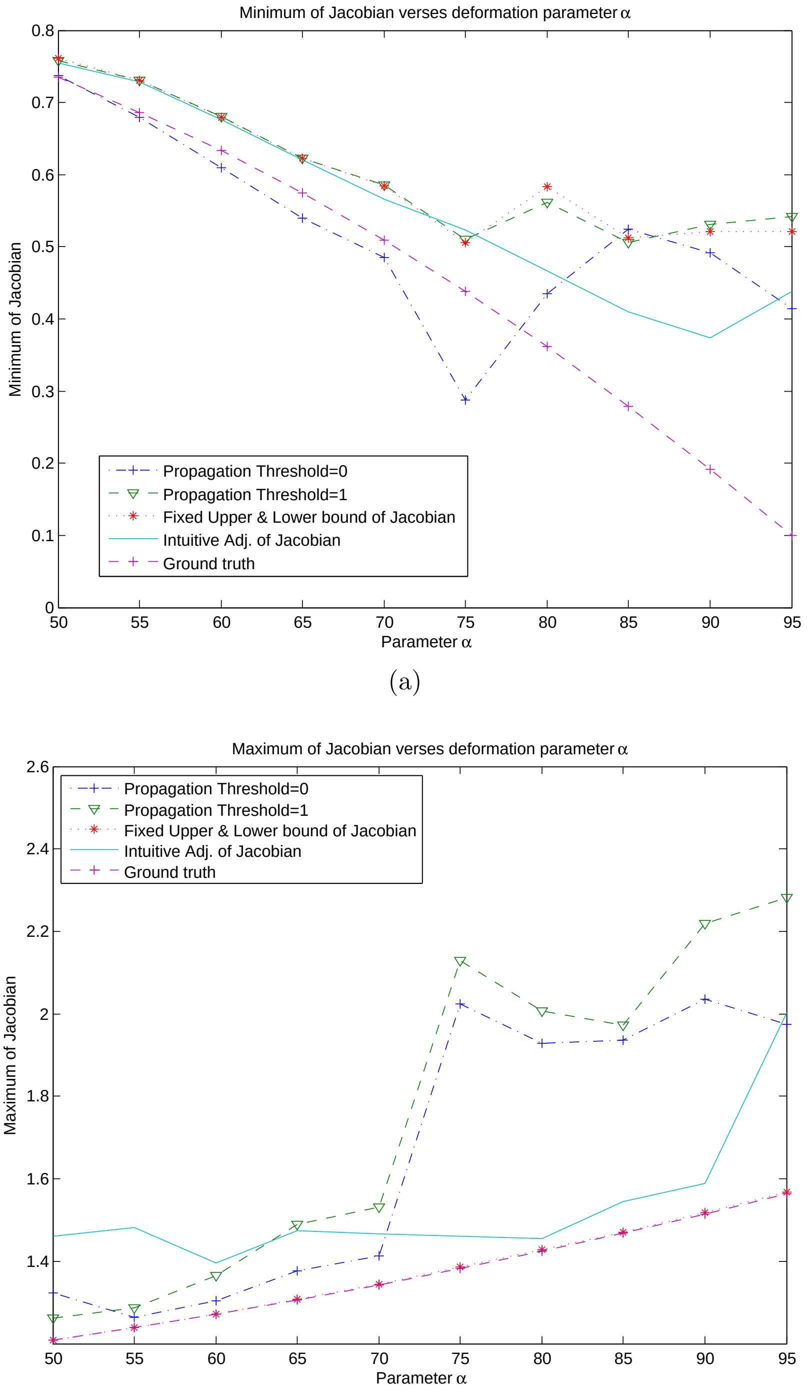
Propagation Threshold:	0	1	Fixed Upper and Lower Bound of Jacobian	Intuitive Adjustment of Jacobian
Max. deformation paprameter α	75	77	88	93
Average SSD	41.13356	34.45255	46.58572	47.48816
Mean Warping Index	0.32177	0.2967	0.13254	0.15808
Max. Warping Index	1.88091	1.64602	0.70829	0.79352
Average number of iteration	858.9375	986.82353	1000	1000



(a)

(b)

Figure 4.47. Experimental results by using demons algorithm. $\sigma = 3$, Gaussian filter size=63(half of image size) (a)Mean warping index (b)Maximum warping index.



(a)

(b)

Figure 4.48. Experimental results by using demons algorithm. $\sigma = 3$, Gaussian filter size=63(half of image size) (a)Minimum of the Jacobian determinant (b)Maximum of the Jacobian determinant.

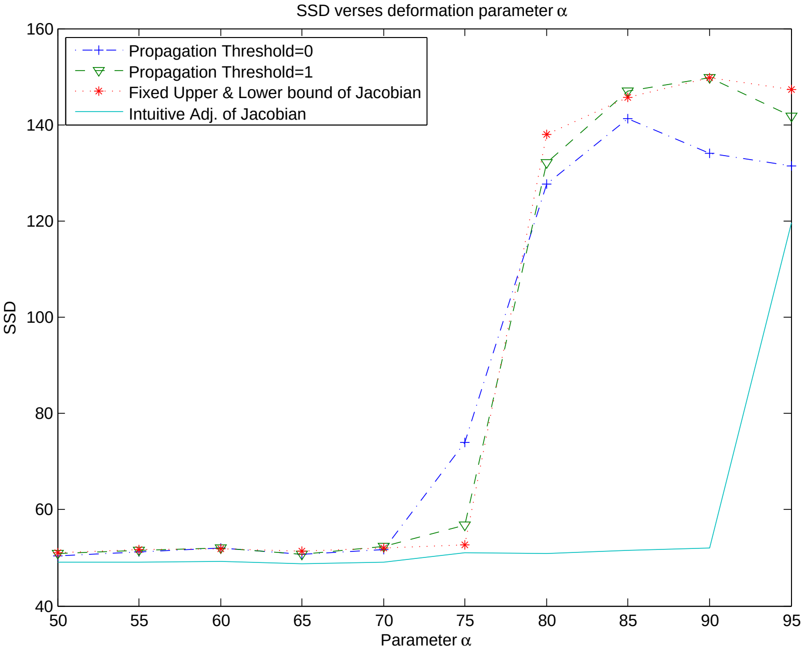


Figure 4.49. SSD measurement of demons algorithm. $\sigma = 3$, Gaussian filter size=63(half of image size).

Table 4.6. Summary of demons experiments on synthetic data with ground truth

Propagation Threshold:	0	1	Fixed Upper and Lower Bound of Jacobian	Intuitive Adjustment of Jacobian
Max. deformation paprameter α	70	70	75	90
Average SSD	51.20649	51.50083	51.78674	50.10526
Mean Warping Index	0.17262	0.17832	0.17851	0.17806
Max. Warping Index	0.74787	1.00727	1.04062	0.89841
Average number of iteration	230	216	258.5	617.4

These plots display SSD values, mean warping index values and maximum warping index values verse the values of the deformation parameter α for each registration algorithms. We focus on the results of maximum warping indexes to compare the accuracy of different regirdding methods. The maximum iteration and the determinant of Jacobian are used to set the stop criteria in our experiments. Here we set the maximum iteration to 1000 and the determinant of Jacobian should always be positive.

4.2.1 Original Method

The results of the original elastic registration is not compared in the experiment due to the use of a large regularization weight to ensure the convergence of the algorithm. If a smaller regularization parameter is used, the results suffer from being divergent. It is also mentioned in [22,32] that the elastic registration is not suitable for large-deformation applications. Fluid registration algorithm correctly converged until the deformation parameter $\alpha = 75$ as shown in Fig. 4.38 and Table. 4.3. Diffusion, curvature, and demons registration algorithms correctly converged until $\alpha = 72, 75$, and 70 respectively.

4.2.2 Maximum Regridding Frequency

With maximum regridding frequency, diffusion registration and curvature registration algorithms outperform (α=74, 77) the original ones (α=72, 75) in terms of robustness and accuracy as shown in Fig. 4.41, Table 4.4, Fig. 4.44, and Table. 4.5. The improvement of fluid and demons registration algorithms is not obviously (Fig. 4.38, Table 4.3, Fig. 4.47, and Table 4.6) by our experiments. But all the figures show that the quality of registration is better or similar to the original methods with the maximum regridding frequency.

Since the maximum regridding frequency is required for our proposed method, it is still feasible by using maximum regridding frequency in all the registration algorithms.

4.2.3 Fixed Upper and Lower Bounds of Jacobian

It is natural to explore the behavior of the registration results if the Jacobian determinant values are restricted within the range of the actual values. This is accomplished by setting the parameters J_{min} and J_{max} in Algorithm 4 to the actual minimal and maximum Jacobian determinant values of the transformations.

From Fig. 4.35, Fig. 4.38, Fig. 4.41, Fig. 4.44, and Fig. 4.47, we observe that the occurrence of abrupt change of each registration algorithms is deferred (α=80, 80, 84, 88, and 75 respectively). It indicates that by confining the Jacobian determinant of the transformation to the actual range, the robustness of the registration algorithm can be improved.

4.2.4 Intuitive Adjustment of Jacobian

The plots denote as "Intuitive Adjustment of Jacobian" are the results of setting the limitation by gradually relaxing upper and lower bounds of the Jacobian determinant (Algorithm 5). Surprisingly, compared to the method using the actual range for the Jacobian determinant values, the "Intuitive Adjustment of Jacobian" method shows a dramatic improvement of each registration algorithms. Elastic, fluid, diffusion, curvature and demons are able to register the image pair till the deformation parameter α=85, 85, 86, 93, and 90. In addition, the average and maximum warping indices are generally lower.

This indicates that gradually relaxing the upper and lower bounds of the Jacobian determinant is a practical strategy to compensate the lack of the prior knowledge about the range of the Jacobian determinant values.

4.2.5 Jacobian Determinant

It is also interesting to show the minimum and maximum Jacobian determinant values of the recovered transformations in our experiments. From Fig. 4.36, Fig. 4.39, Fig. 4.42, Fig. 4.45, and Fig. 4.48, we observe that there seems a correlation between the locations of the abrupt changes of the curves showing the maximal Jacobian determinant values and those shown in Fig. 4.35, Fig. 4.38, Fig. 4.41, Fig. 4.44 and Fig. 4.47. It implies that the algorithms fail to converge correctly when the maximum Jacobian determinant values significantly deviate from the actual value.

CHAPTER 5

CONCLUSION

In this dissertation, we presented a regridding scheme with a novel grid repairing mechanism. Regridding frequency is maximized without causing any accumulated error by always resampling the original template instead of the latest propagated template. The proposed regridding scheme can be incorporated into many existing nonrigid image registration algorithms as shown in this dissertation. A general flow chart was also provided. Specifically, we demonstrate how the proposed regridding scheme is integrated into the elastic, fluid, diffusion, curvature and demons registration algorithms and makes them powerful registration algorithms for large-deformation applications.

The performance was compared to the original method and two regridding methods for each registration algorithm. Our experiments showed that with the proposed regrididng scheme integrated, all registration algorithms ca onutperform the original ones in terms of robustness and accuracy.

Another possible usage of the proposed grid repairing method is the construction of local volume preserving nonrigid registration algorithms [33–36]. This can be achieved by setting both the parameters J_{min} and J_{max} to be close to 1 and will be further investigated in the future.

REFERENCES

[1] J. Maintz and M. Viergever, “An overview of medical image registration methods,” 1996.

[2] C. Sorzano, P. Thevenaz, and M. Unser, “Elastic registration of biological, images using vector-spline regularization,” *Ieee Transactions On Biomedical Engineering*, vol. 52, no. 4, pp. 652–663, APR 2005.

[3] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, “Mutual-information-based registration of medical images: a survey,” *IEEE Transactions on Medical Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[4] A. Guimond, A. Roche, N. Ayache, and J. Meunier, “Three-dimensional multimodal brain warping using the demons algorithm and adaptive intensity corrections,” *Ieee Transactions On Medical Imaging*, vol. 20, no. 1, pp. 58–69, JAN 2001.

[5] H. mei Chen, P. K. Varshney, J. Luo, and T. hung Lin, “A global optimization scheme for mutual information based remote sensing image registration,” in *Advanced Concepts for Intelligent Vision Systems*, Brussels, Belgium, 2004, pp. 349–356.

[6] H. mei Chen, M. K. Arora, and P. K. Varshney, “Mutual information based image registration for remote sensing data,” *International Journal of Remote Sensing*, vol. 24, no. 18, pp. 3701–3706, September 2003.

[7] H.-M. Chen, P. Varshney, and M. Arora, “Performance of mutual information similarity measure for registration of multitemporal remote sensing images,”

IEEE Transactions on Geoscience and Remote Sensing, vol. 41, no. 11, pp. 2445–2454, November 2003.

[8] J. Ashburner, C. Hutton, R. Frackowiak, I. Johnsrude, C. Price, and K. Friston, "Identifying global anatomical differences: deformationbased morphometry," *Hum. Brain Map*, vol. 6, pp. 348–357, 1998.

[9] J. Modersitzki and B. Fischer, "Fast diffusion registration," *AMS Contemporary Mathematics, Inverse Problems, Image Analysis, and Medical Imaging*, vol. 313, pp. 117–129, 2002.

[10] B. Fischer and J. Modersitzki, "A unified approach to fast image registration and a new curvature based registration technique," *Linear Algebra and its Application*, vol. 308, pp. 107–24, 2004.

[11] 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.

[12] G. Liao and D. Anderson, "A new approach to grid generation," *Applicable Analysis*, vol. 44, no. 3, pp. 285–298, 1992.

[13] G. Liao, T. Pan, and J. Su, "A numerical grid generator based on Moser's deformation method," *Numerical Methods for Partial Differential Equations*, vol. 10, no. 1, pp. 21–31, 1994.

[14] G. Liao and J. Su, "Grid generation via deformation," *Applied Mathematics Letters*, vol. 5, no. 3, pp. 27–29, 1992.

[15] J.-P. Thirion, "Image matching as a diffusion process: an analogy with Maxwell's demons," *Medical Image Analysis*, vol. 2, no. 3, pp. 243–260, 1998.

[16] J. Cao and K. J. Worsley, "The geometry of the hotelling's random field with applications to the detection of shape changes," *Ann. Stat*, 1998.

[17] C. Gaser, “Detecting structural changes in whole brain based on nonlinear deformations—application to schizophrenia research,” *NeuroImage*, vol. 10, pp. 107–113, 1999.

[18] N. Andreasen, S. Arndt, n. Swayze, V, T. Cizadlo, M. Flaum, D. O’Leary, J. Ehrhardt, and W. Yuh, “Thalamic abnormalities in schizophrenia visualized through magnetic resonance image averaging,” *Science*, vol. 266, no. 5183, pp. 294–298, 1994.

[19] J. Cao and K. Worsley, “The geometry of correlation fields with an application to functional connectivity of the brain,” *Ann. Appl. Probab.*, vol. 9, no. 4, pp. 1021–1057, 1999.

[20] K. Chen, E. M. Reiman, G. E. Alexander, D. Bandy, R. Renaut, W. R. Crum, N. C. Fox, and M. N. Rosso, “An automated algorithm for the computation of brain volume change from sequential mris using an iterative principal component analysis and its evaluation for the assessment of whole-brain atrophy rates in patients with probable alzheimer’s disease,” *NeuroImage*, vol. 22, no. 1, pp. 134–143, 5 2004.

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

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

[23] R. Monedero, J. Ruiz, and J. Sanchez, “Frequency implementation of the euler-lagrange equations for variational image registration,” *Signal Processing Letters, IEEE*, vol. 15, pp. 321–324, 2008.

[24] H. K. P. Bkp and S. G. Bg, “Determining optical flow,” *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[25] B. Jiag, *The Least-Squares Finite Element Method.* Springer, 1998.

[26] J. Demmel, “Solving the Discrete Poisson Equation using Jacobi, SOR, Conjugate Gradients, and the FFT,” March 1996. [Online]. Available: http://www.cs.berkeley.edu/ demmel/cs267/lecture24/lecture24.html

[27] ——, “Solving the Discrete Poisson Equation using Multigrid,” March 1996. [Online]. Available: http://www.cs.berkeley.edu/ demmel/cs267/lecture25/lecture25.html

[28] G. E. Forsythe, M. A. Malcolm, and C. B. Moler, *Computer Methods for Mathematical Computations*. Prentice-Hall, 1977.

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

[30] National Library of Medicine (U.S.). Board of Regents, *Electronic imaging : report of the Board of Regents*, ser. NIH publication ; no. 90-2197. Bethesda, Md. : U.S. Dept. of Health and Human Services, Public Health Service, National Institutes of Health, 1990.

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

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

[33] I. Yanovsky, P. M. Thompson, S. Osher, and A. D. Leow, “Topology preserving log-unbiased nonlinear image registration: Theory and implementation.” in *CVPR*. IEEE Computer Society, 2007.

[34] T. Rohlfing, C. R. M. Jr., D. A. Bluemke, and M. A. Jacobs, “Volume-preserving non-rigid registration of MR breast images using free-form deformation with an incompressibility constraint,” *IEEE Trans. Med. Imaging*, vol. 22, no. 6, pp. 730–741, 2003.

[35] E. Haber and J. Modersitzki, "Numerical methods for volume preserving image registration," in *MICCAI (1)*, 2004, pp. 591–598.

[36] ——, "A scale space method for volume preserving image registration," in *Scale-Space*, 2005, pp. 561–572.

BIOGRAPHICAL INFORMATION

Ting-Hung Lin was born on December 31, 1976 in Taipei, Taiwan. He received his Bachelor of Science degree in Computer Science from Tunghai University in Taiwan in June 1999 where he focused on the field of Computer Network. He began his graduate studies in Computer Science and Engineering at the University of Texas at Arlington in the fall of 2002 and received the Graduate Dean Fellowship for two years. While study at the University of Texas at Arlington he emphasized in the area of digital image processing and computer vision. He received his Master of Science in Computer Science and Engineering from the University of Texas at Arlington in December 2004. Ting-Hung Lin continued the education toward the doctoral program in Computer Science and Engineering at the University of Texas at Arlington in January 2005. And he received his PhD degree of Science in Computer Science and Engineering from the University of Texas at Arlington in December 2008. His research interests include image registration, optimization, digital image processing, computer vision and visualization with applications in medical imaging and remote sensing.