

(c) 2002 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

# Computational Cost of Non-Rigid Registration Algorithms based on Fluid Dynamics

Gert Wollny and Frithjof Kruggel

*Abstract*—Though fluid dynamics offer a good approach to non-rigid registration and give accurate results, even with large scale deformations, its application is still very time consuming. We introduce and discuss different approaches to solve the core problem of non-rigid registration, the partial differential equation of fluid dynamics. We focus on the solvers, their computational costs and the accuracy of registration. Numerical experiments show that relaxation is currently the best approach, especially when reducing the cost per iteration by focusing the updates on deformation spots.

*Keywords*—registration, fluid dynamics, computational cost, medical images

## I. INTRODUCTION

Non-rigid image registration is a problem in medical computer vision that has many applications. For instance, non-rigid registration can be used to detect changes in MR time series images, arising when pathological processes (e.g. tumor growth, scarification, and atrophies) are monitored by a temporal series of MR examinations. Another application is non-rigid image-atlas registration that is used to locate brain-structures in the images.

Registration is achieved by applying one transformation to one image (which we call the study image), in order to match another (reference) image with respect to a given cost function. In practice, these transformations must accommodate both very complex and large deformations.

Early approaches of non-rigid registration, first introduced by Baiscy et al. [1], [2], later by Evans et al. [3], Miller et al. [4], and Christensen et al. [5], were all based on linear elasticity. As linear elasticity restricts the registration to globally smooth and therefore to locally small deformations, these methods fail to achieve very complex and/or large deformations. In an extension to their initial work, Christensen et al. [6] described a registration approach in which a viscous fluid model was used to control the deformation. In particular, the study image is modeled as a viscous fluid which is able to flow so as to match the reference. In this model internal forces disappear gradually, due to the attenuation in viscous fluids. This way, the desired deformation can be fully achieved, even if large scale deformations are required. Other approaches to non-rigid registration are discussed in the literature (cf. e.g. [7], [8], [9]), but will not be considered in this paper.

The original implementation of fluid dynamics based registration, based on *successive over-relaxation* (SOR), is demanding with respect to (w.r.t.) computational cost and is thus time consuming. Bro-Nielsen et al. [10] proposed a further algorithm, based on *convolution filters* (CONV). They suggest that its performance results in a speedup of at least an order of magnitude.

In this paper, we will compare the original algorithm of Christensen [6], to the convolution based algorithm of Bro-Nielsen

[10], and to an alternative, which is based on the *minimum residual algorithm* (MINRES), in terms of speed, memory usage, and registration accuracy. We will study optimization by employing an adaptive update scheme, which focuses on regions with significant deformations. Finally, we will demonstrate that it is possible to achieve good registration results for high resolution MRI brain image data sets, in a reasonable amount of time on workstation class computers.

## II. REGISTRATION BASED ON FLUID DYNAMICS

If not indicated otherwise, all numbers, vectors, matrices, and functions considered in this paper are real. Vectors are denoted as bold lower case characters, such as  $\mathbf{x}$ , and  $\mathbf{v}$ , matrices appear as bold face upper case letters.

An image  $I$  is defined as a mapping  $I : \Omega \rightarrow V$ , with  $\Omega := [0, 1]^3 \subset \mathbb{R}^3$  being the image domain and  $V \subset \mathbb{R}$  being the (intensity-) range. The ordered pair  $(\mathbf{x}, I(\mathbf{x})) \in \Omega \times V$  of a coordinate, and its corresponding intensity value is called a voxel (volume element). In the following  $R : \Omega \rightarrow V$  denotes the reference and  $S : \Omega \rightarrow V$  the study image.

The transformation  $T : \Omega \rightarrow \Omega$  of an image  $I$  is a mapping of the image domain to itself, and  $\Gamma$  is the set of all such transformations. Following Christensen et al. [6], we use an *Eulerian reference frame* to describe a transformation  $T$ . Here, voxels of the deforming image are tracked by their position. Given a vector field  $\mathbf{u}$ , and a voxel originating at  $\mathbf{x}$ , at time  $t_0 := 0$ . With  $\mathbf{u}(\mathbf{x}, t)$  being the displacement of this voxel at time  $t$ , its new location will be  $\mathbf{x} - \mathbf{u}(\mathbf{x}, t)$ . Hence, transformation  $T$  is defined as  $T(\mathbf{x}, t) := \mathbf{x} - \mathbf{u}(\mathbf{x}, t)$ . The concatenation of two transformations  $T_1$  and  $T_2$  with the displacement fields  $\mathbf{u}_1$  and  $\mathbf{u}_2$  is given by:

$$T_1 \circ T_2 := \mathbf{x} - \mathbf{u}_1(\mathbf{x} - \mathbf{u}_2(\mathbf{x})) - \mathbf{u}_2(\mathbf{x}). \quad (1)$$

Additionally, the velocity field in an Eulerian reference frame is determined by [6]:

$$\mathbf{v}(\mathbf{x}, t) = \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} + \nabla \mathbf{u}(\mathbf{x}, t) \mathbf{v}(\mathbf{x}, t), \quad (2)$$

with the latter term  $\nabla \mathbf{u}(\mathbf{x}, t) \mathbf{v}(\mathbf{x}, t)$  accounting for the kinematic non-linearity of the voxels.

The purpose of registration is to find a transformation  $T_{\min}$ , which minimizes a given cost function  $F_c(T, S, R)$  in conjunction with some energy normalization (or smoothness measure)  $E(T)$ :

$$T_{\min} := \arg \min_{T \in \Gamma} (F_c(T, S, R) + \kappa E(T)), \quad (3)$$

with  $\kappa$  being a Lagrangian multiplier to balance registration accuracy and transformation smoothness.

Solving this minimization problem (3), can be done by finding the zero point of its first order derivative

$$\kappa \frac{\partial}{\partial T} E(T) = - \frac{\partial}{\partial T} F(T, S, R). \quad (4)$$

As we study synthetically generated transformations only, the study image as well as the reference image will have the same intensity distribution. Hence, we use the simplest case of a similarity measure<sup>1</sup>, the so-called *identity relationship* [15], which yields the sum of squared intensity differences as a cost function:

$$F_c(T, S, R) := \frac{1}{2} \int_{\Omega} [R(\mathbf{x}) - S(T(\mathbf{x}))]^2 d\mathbf{x}. \quad (5)$$

Its first order derivative is then used as the force  $\mathbf{f}$  to drive the registration

$$\begin{aligned} \mathbf{f}(\mathbf{x}, T(\mathbf{x}, t)) &:= \left. \frac{\partial}{\partial T} F(T, S, R) \right|_{\mathbf{x}} \\ &= [S(T(\mathbf{x}, t)) - R(\mathbf{x})] \nabla S|_{T(\mathbf{x}, t)}. \end{aligned} \quad (6)$$

Using fluid dynamics as the energy regularisation, and with  $\kappa := 0$  (4) finally reads

$$(\mu \nabla^2 + (\mu + \lambda) \nabla(\nabla \cdot)) \mathbf{v}(\mathbf{x}, t) = -\mathbf{f}(\mathbf{x}, \mathbf{u}(\mathbf{x}, t)), \quad (7)$$

with  $\mu$  and  $\lambda$  being Lamé's elasticity constants [16]. The term  $\mu \nabla^2 + (\mu + \lambda) \nabla(\nabla \cdot)$  is the Navier-Stokes operator, modified to account for non-mass-conserving deformations [17].

Solving the registration problem is done iteratively over a time step of  $\Delta t$ . In each step, (7) is solved for a constant time to estimate the current velocity field  $\mathbf{v}$ , which is then used to update the displacement field  $\mathbf{u}$  by a time integration step derived from (2):

$$\begin{aligned} \mathbf{u}(\mathbf{x}, t_{i+1}) &:= \mathbf{u}(\mathbf{x}, t_i) + \\ &\Delta t [\mathbf{v}(\mathbf{x}, t_i) - \nabla \mathbf{u}(\mathbf{x}, t_i) \mathbf{v}(\mathbf{x}, t_i)]. \end{aligned} \quad (8)$$

To solve the registration problem, the continuous image domain  $\Omega$  is discretized by using a grid constant  $g > 1, g \in \mathbb{N}$

$$\Omega \rightarrow \hat{\Omega} := \left\{ \frac{1}{g-1} \begin{pmatrix} i \\ j \\ k \end{pmatrix} \middle| i, j, k = 0, 1, \dots, g-1 \right\}. \quad (9)$$

With  $n := g^3$ , the discretization of a vector field  $\mathbf{x}$  on the domain  $\hat{\Omega} \subset \mathbb{R}^3$  corresponds to a  $n$ -dimensional vector  $[\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{n-1}]^T$ . Given  $m := i + jg + kg^2$  ( $i, j, k \in \hat{\Omega}$ ), the  $m$ th component  $\mathbf{x}_m$  is also denoted as  $\mathbf{x}_{i,j,k}$ .

Since we work on a discretized image domain, the transformation  $T$  is not preserving topology, per se. Therefore, we have to keep the minimal value of the Jacobian of transformation  $T$  [18]

$$J(T) := \min_{\mathbf{x} \in \Omega} \det(\mathbf{I} - \nabla \mathbf{u}(\mathbf{x}, t)), \quad (10)$$

<sup>1</sup>For a detailed discussion of similarity measures see [11], [12], [13], [14]

from falling below a certain threshold. Christensen et al. [6] achieve this by re-gridding: Every time  $J(T)$  drops below the heuristic value 0.5, the global deformation  $T$  is updated  $T := T \circ T_t$  by using the current transformation  $T_t := \mathbf{x} - \mathbf{u}(\mathbf{x}, t)$ , and a new template  $\hat{S}$  is generated by applying the current transformation  $\hat{S} := S(T)$ . The displacement field  $\mathbf{u}$  and time  $t$  are set zero, and for further registration the new template  $\hat{S}$  is used.

To avoid local minima and to speed up computation, a coarse-to-fine multi-resolution scheme can be employed voluntarily: We start with a coarse discretization of the image domain  $\Omega$ , introduced by a low grid constant  $g$ . If registration is achieved at a certain grid level, the grid constant will be increased. The obtained transformation is tri-linearly interpolated on the higher resolution, thus assuming that the minimal value of the transformation's Jacobian is still positive. The multi-resolution iteration is stopped when the grid constant  $g$  is as large as the finite resolution of the input images. For implementation details of the registration algorithm, please refer to the software we made available online under the terms of the GNU public license [19].

To compare the results of registration approaches, finally a measure for registration accuracy has to be introduced. Since the synthetical transformation, used to deform the images, usually will not minimize the energy regularisation term, comparing this transformation to the obtained registration transformation is not an option. Instead, we propose an accuracy measure based on the relative value of the cost function: With identity mapping  $T_0(\mathbf{x}) := \mathbf{x} \forall \mathbf{x} \in \Omega$ , the accuracy of registration achieved by the transformation  $T(\mathbf{x}) : \Omega \rightarrow \Omega$  is defined as:

$$A_{\text{reg}}(T) := \frac{F_c(T_0, S, R)}{F_c(T, S, R)}. \quad (11)$$

Given a perfect registration, the accuracy becomes infinite. For interpolation of the image data during registration introduces errors in the image data, we do not expect infinite registration accuracy in our experiments.

### III. APPROACHES FOR SOLVING THE LINEAR PDE

The time consuming step of this registration algorithm, and therefore its core problem, is the solution of PDE (7) for a constant time and force. We will now discuss different approaches for its solution on the discretized coordinate domain  $\hat{\Omega}$ , namely (A): successive over-relaxation; (B): successive over-relaxation with adaptive update; (C): the minimal residual algorithm; and (D): Convolution Filters. We will compare the approaches in terms of speed, memory usage, and accuracy of the resulting registration.

#### A. Successive Over-Relaxation (SOR)

Discretizing (7), using finite differences [20], yields a linear system

$$\mathbf{A} \mathbf{v} = \mathbf{f}, \mathbf{A} \in \mathbb{R}^{3n \times 3n}, \mathbf{v}, \mathbf{f} \in \mathbb{R}^{3n}. \quad (12)$$

One method to solve (12) is *successive over-relaxation* (SOR) [21], [22], [23], [20]. Splitting  $A = (a_{r,s}) \in \mathbb{R}^{3n \times 3n}$  into a diagonal matrix  $\mathbf{A}_d$ , a lower left matrix  $\mathbf{A}_l$ , and an upper right one  $\mathbf{A}_r$ ,

$$\mathbf{A} = \mathbf{A}_d + \mathbf{A}_l + \mathbf{A}_r; \quad (13)$$

we obtain the iteration rule of SOR, with the iteration index  $l$  and the over-relaxation factor  $\omega$ :

$$\mathbf{v}^{(l+1)} := \mathbf{v}^{(l)} + \omega \mathbf{A}_d^{-1} \left( \mathbf{f} - \mathbf{A}_l \mathbf{v}^{(l+1)} - (\mathbf{I} + \mathbf{A}_r) \mathbf{v}^{(l)} \right). \quad (14)$$

In each iteration, the component  $\mathbf{v}_m$  of vector field  $\mathbf{v}$  is updated with a residual

$$\mathbf{r}_m^{(l+1)} := \frac{\omega}{a_{mm}} \left( \mathbf{f}_m - \mathbf{v}_m^l - \sum_{j=1}^{m-1} a_{m,j} \mathbf{v}_j^{(l+1)} - \sum_{j=m+1}^n a_{m,j} \mathbf{v}_j^{(l)} \right). \quad (15)$$

Note, that in the term  $\sum_{j=1}^{m-1} a_{m,j} \mathbf{v}_j^{(l+1)}$   $j < m$ , i.e. to evaluate  $\mathbf{r}_m^{(l+1)}$ , already updated elements of  $\mathbf{v}$  are used.

The sparse structure of  $\mathbf{A}$  yields that one update of the discretized velocity field  $\mathbf{v}$  needs an order of  $66n$  floating point operations ( $O(66n)$  FLOPs) [19]. The update scheme does not require additional memory cells for the iteration process.

### B. Successive Over-Relaxation with Adaptive Update

Non-rigid registration is usually preceded by rigid registration. Thus, we only expect differences between the images in certain regions of interest. Since we do a coarse-to-fine multi-resolution processing, large differences are roughly registered at coarser levels, yet. Hence, large regions of the images may have little (if any) influence on the solution of PDE (7).

By deriving the residuum  $\mathbf{r}_{i,j,k} = \mathbf{r}_m$  (15), in each iteration  $\mathbf{v}_{i,j,k}$  depends only on the 19 values with indices  $\beta \in \mathfrak{S}$ :

$$\mathfrak{S} := \left\{ \begin{pmatrix} i \\ j \\ k \end{pmatrix}, \begin{pmatrix} i \pm 1 \\ j \\ k \end{pmatrix}, \begin{pmatrix} i \\ j \pm 1 \\ k \end{pmatrix}, \begin{pmatrix} i \\ j \\ k \pm 1 \end{pmatrix}, \begin{pmatrix} i \pm 1 \\ j \pm 1 \\ k \end{pmatrix}, \begin{pmatrix} i \\ j \pm 1 \\ k \pm 1 \end{pmatrix}, \begin{pmatrix} i \pm 1 \\ j \\ k \pm 1 \end{pmatrix} \right\}. \quad (16)$$

Therefore, it makes sense to update  $\mathbf{v}_{i,j,k}$  only if at least one residuum  $\|\mathbf{r}_{\beta \in \mathfrak{S}}\|$  is larger than a threshold value  $\hat{r}$  evaluated in the preceding iteration step:

$$\hat{r} := \begin{cases} 0 & m = 1 \\ \bar{r}^{(m)} \cdot \frac{\bar{r}^{(m)}}{\bar{r}^{(m-1)}} \cdot \frac{1}{m^2} & otherwise \end{cases}, \quad (17)$$

with

$$\bar{r} := \frac{1}{X \cdot Y \cdot Z} \sqrt{\sum \|\mathbf{r}_{i,j,k}\|^2}, \quad (18)$$

To obtain all residues  $\mathbf{r}_{i,j,k}$  initially, this threshold  $\hat{r}$  is set to zero during the first iteration. Later on, the threshold is set to the average over the square norms  $\bar{r}$  of the residual vectors combined with a term which involves the evolution of the respective residuum and another term to decrease the threshold in subsequent iterations, generally. With an increasing number

of iterations, the solution search space thus increases and converges to the search space of the original SOR. Hence, if SOR is convergent, then SOR with adaptive update (SORA) will also converge. This update scheme (SORA) can also be seen as a variant of the Gauss-Southwell-Relaxation [24].

The number of operations, needed for one update of the velocity field, depends on the input data, but is well below the  $O(66n)$  FLOPs that are needed for the unmodified SOR. Additional storage  $O(n)$  is required for residues and update markers.

### C. The Minimum Residual Algorithm (MINRES)

The solution of the linear system (12) can also be regarded as the minimum of:

$$\Phi(\mathbf{v}) := \frac{1}{2} (\mathbf{A}\mathbf{v} - \mathbf{f})^2. \quad (19)$$

If  $\mathbf{A}$  is a symmetric matrix, though not necessarily a positive definite one, the *minimum residual algorithm*<sup>2</sup> [21], as a variant of the conjugate gradient method, can be employed to solve this minimization problem.

In the linear system (12), the diagonal elements of  $\mathbf{A}$  are  $a_{i,i} = 1$ , thus the system can be considered as treated with a Jacobi pre-conditioner. For the simple structure, a multiplication with  $\mathbf{A}$  is achieved with only  $O(51n)$  FLOPs [19]. Hence, solving (12) by using MINRES seems feasible. In summary, this algorithm requires  $O(117n)$  FLOPs per iteration [19]. No storage for  $\mathbf{A}$  is required, but the algorithm requires  $O(n)$  storage cells for temporary vectors. Employing an adaptive update, similar to the one given for SORA, would break the search strategy of MINRES, which depends on orthogonal search directions. Thus, an adaptive update was not considered.

### D. Convolution filters

Bro-Nielsen et al. [10] suggested the usage of convolution filters for solving (7). A linear operator comprising the form of (7) is given as:

$$\Lambda := \mu \nabla^2 + (\mu + \lambda) \nabla (\nabla \cdot). \quad (20)$$

With a filter width parameter  $d$ , the filter components  $\Theta \in \mathbb{R}^{3 \times 3}$ , as given by Bro-Nielsen [25], and  $\mathbf{y}_{r,s,t} = (\frac{r}{d}, \frac{s}{d}, \frac{t}{d})^T$  |  $r, s, t \in [-d, d] \cap \mathbb{Z}$ , the solution of PDE (7) at  $\mathbf{x} \in \Omega$  can be calculated by convoluting the impulse response of operator  $\Lambda$  with the input force  $\mathbf{f}$ :

$$\mathbf{v}(\mathbf{x}) = \sum_{r,s,t=-d}^d \Theta(\mathbf{y}_{r,s,t}) \cdot \mathbf{f}(\mathbf{x} - \mathbf{y}_{r,s,t}). \quad (21)$$

The filter components  $\Theta \in \mathbb{R}^{3 \times 3}$  can be pre-calculated. For each location  $(r,s,t)$  in (21) a product with a  $3 \times 3$  matrix and the sum of two 3D vectors have to be calculated. The computational cost for solving (7) on the discretized grid  $\hat{\Omega}$  is  $O(18(2d+1)^3 n)$  FLOPs, therefore. Additional storage space is only needed for the filter components; it can be neglected, because the filter

<sup>2</sup>The usage of the standard conjugated gradient method (CG) is discouraged; without any boundary conditions,  $\mathbf{A}$  is singular and thus not positive definite. Even with boundary conditions specified, for large  $n$   $\mathbf{A}$  becomes ill conditioned, which will lead to an instable operation of CG.

width is small compared to the image size. A given filter width restricts the maximum length of the deformation during a transformation step in the registration algorithm. For a filter width smaller than the largest image dimension, it is not possible to obtain an accurate solution of PDE (7). Thus, a multi-resolution approach for solving the registration problem is mandatory to accommodate large deformations [10].

We now compare approaches A-D w.r.t, the respective computational costs, as obtained above (Table I).

[Insert Table I approximately here]

MINRES needs nearly twice as many FLOPs per iteration as SOR does. Since CG-based methods are known for their fast convergence, MINRES may perform better. The cost of CONV increases very quickly with a growing filter width.

To get a first impression of the performance ratio between CONV and iterative methods, we additionally summarize the number of iterations, with which the theoretical computational cost of these methods exceed the ones of the convolution based solver (Table II), for different filter sizes.

[Insert Table II approximately here]

These numbers indicate, that using CONV with large filter sizes does not offer an advantage compared to iterative solvers, if the number of iterations can somehow be restricted with these methods. On the other hand, we have to consider which filter sizes are sufficient to solve (7) with an adequate precision to achieve a good registration.

#### IV. EXPERIMENTS AND RESULTS

Convergence properties of solvers for systems of linear equations, such as SOR, SORA and MINRES, are well known from literature (e.g. [20]). Nevertheless, we first ran a test to determine the performance of the iteration methods alone, i.e. outside the context of the registration problem, since the actual convergence properties are also problem specific. Discrete random force fields  $\mathbf{f}$  on a grid of  $128 \times 128 \times 128$  voxels were generated, and (7) was solved. Every time the relative residuum of the solution dropped below a certain threshold during the iteration, time and number of iterations were measured. If matrix  $\mathbf{A}$  is a symmetric, positive definite, tridiagonal, or block wise tridiagonal matrix, the theoretically optimal relaxation factor  $\omega$  can be approximated by:

$$\omega = \frac{2}{1 + \sqrt{1 - \lambda^2}} \quad (22)$$

with  $\lambda$  being the *spectral radius* of  $\mathbf{A}_d^{-1}(\mathbf{A}_l + \mathbf{A}_r)$  [20]. Since it is difficult to calculate the spectral radius, we used relaxation factors of 0.8, 1.0, 1.4, and 1.8. A dynamic adaption scheme, known as Chebyshev acceleration [20], was also examined, but did not offer any improvement.

The top chart of Fig. 1 shows<sup>3</sup> that the minimal residual method (MINRES) requires least iterations to achieve a given relative residuum.

[Insert Fig.1 approximately here]

<sup>3</sup>Relaxation factors of 0.8 and 1.8 did not yield superior results over factors 1.0 and 1.4 and were omitted from figures.

In terms of computation time, (using a 450Mhz Pentium II based workstation), however, successive over-relaxation was faster (see bottom chart of Fig. 1).

In the generated fields, forces are distributed evenly over the domain, therefore the adaptive update showed a slight advantage. We expect better results on realistic data, because forces will typically be localized. An over-relaxation factor of 1.0 achieved best results for the relaxation based methods, and it was used in the following registration experiments. On the basis of this experiment, we expect SOR or SORA to perform better than MINRES.

The same test was run with convolution filters. Especially by using small filter widths, results for the solution of PDE (7), were not satisfactory (Table III).

[insert table III approximately here]

This conforms with the statement that CONV does not yield a proper solution of (7), if the filter width is much below the image dimensions. Filters of higher order were not tested, since their high computational costs do not offer an advantage compared to the iterative methods.

For our second experiment, the performance of the different methods was tested in the framework of the registration algorithm by using synthetic images of size  $64 \times 64 \times 64$  (see also Fig. 2).

[Insert Fig. 2 approximately here]

Twenty reference images were generated, deforming the study with a smooth random deformation field, and the study was registered to these references. We choose  $\mu = \lambda = 1.0$  and the multi-resolution start grid constant  $g = 16$ .

[Insert Table IV approximately here]

Results in Table IV demonstrate, that it is not necessary to solve (7) with a high accuracy. A limit of 10 iterations only, yields a registration quality nearly as good as any higher number of iterations.

[Insert Table V approximately here]

For convolution filters, a minimum filter width of 5 is necessary to achieve results which are similar to those of the other methods (see Tab III and Fig. 2). The bad registration results for a filter width of 3, again show that a small filter is not sufficient to solve (7) with an adequate accuracy. The number of iterations in the iterative methods can be reduced significantly, i.e. it is not necessary to evaluate the velocity field  $\mathbf{v}$  accurately from the deforming force  $\mathbf{f}$  in each iteration step. Thus the computational cost of CONV is similar to those methods, and any speed advantage vanishes. Only with a filter width of 3 convolution filters offers an advantage, in terms of speed but not in terms of accuracy. These results correspond to the ratio of computational costs, as compiled in Table II.

Now, we were interested in testing how algorithms scale on the image resolution, using the optimal parameters as determined in the preceding experiment. For the iteration based methods, we set the maximum number of iterations to 10, or stopped if the relative residuum dropped below 0.01. Convolution filters were applied with a filter width of 5. Although methods for solving (7) need  $O(n)$  FLOPs, the time needed to achieve registration of synthetic images increased at a lower rate than the number of voxels (see Fig. 3).

[Insert Fig. 3 approximately here].

As another conclusion, SOR and SORA scale better than MINRES and CONV. The super-scalar speed increase for small resolutions (i.e.  $32 \times 32 \times 32$ ) is a consequence of caching [26]. SORA proved to be the fastest method.

Finally, we explored the performance of registering high-resolution medical image data. Acceptable results of the registration of pairs of  $250 \times 250 \times 192$  MR image of the head were achieved in approximately 3 hours.

## V. CONCLUSION

We compared two iterative methods *successive over-relaxation* (SOR), and the *minimal residuum algorithm* (MINRES), and a direct approach *convolution filters* (CONV), as methods for the solution of the core problem of non-rigid registration, based on fluid dynamics. In terms of computational costs, we found that MINRES required fewer iterations to solve (7) with a given accuracy, but SOR required less memory cells as well as less time to achieve the same result, and is thus considered superior. Numerical experiments on synthetic data demonstrated the possibility of restricting the numbers of iterations drastically, without a significant loss of accuracy of the registration result. With this optimization, SOR outperformed convolution filters (CONV) in terms of speed while the registration accuracy maintains. Furthermore, we were able to speedup SOR by introducing an adaptive update scheme (SORA). In summary, SORA proved to be the best approach to solve (7).

Although our implementation is neither fully optimized for workstation architectures nor parallelized we were able to obtain results for MRI volumetric data of high resolution, in approximately three hours of computation time. By optimizing our implementation (e.g. by improving the use of the memory cache hierarchies of the processor and by introducing parallelization), a further speedup is expected.

## ACKNOWLEDGMENT

We would like to thank M. Tittgemeyer and N. H. Busch for their helpful discussions, our librarians G. Lewin and A. Hutt for providing the necessary literature, and a special acknowledgment is given to all the creators of free software, since we made extensive use of their work.

## REFERENCES

- [1] R. Bajcsy and S. Kovacic, "Multi-resolution elastic matching," *Computer Vision, Graphics and Image Processing*, vol. 1, no. 46, pp. 1–21, 1989.
- [2] J. C. Gee, M. Reivich, and R. Bajcsy, "Elastically deforming atlas to match anatomical brain images," *Journal of Computer Assisted Tomography*, no. 17, pp. 255–236, 1993.
- [3] A. C. Evans, W. Dai, L. Collins, P. Neelin, and S. Marret, "Warping of a computerized 3-d atlas to match brain image volumes for quantitative neuroanatomical and functional analysis," in *Proc. SPIE Medical Imaging V (1445)*, 1991, pp. 236–246.
- [4] M. I. Miller, G. E. Christensen, Y. Amit, and U. Grenander, "Mathematical textbook of deformable neuroanatomies," in *Proc. Natl. Acad. Sci. USA* 90, 1993, pp. 11944–11948.
- [5] G. E. Christensen, M. I. Miller, and M. Vannier, "A 3d deformable magnetic resonance textbook based on elasticity," in *AAAI Spring Symposium Series: Applications of Computer Vision in Medical Image Processing*. Stanford University, March 1994, pp. 153–156.
- [6] G. E. Christensen, *Deformable shape models for neuroanatomy*, DSc.-thesis, Server Institute of Technology, Washington University, Saint Louis, 1994.
- [7] O. Musse, F. Heitz, and J.-P. Armpach, "Topology preserving deformable image matching using constrained hierarchical parametric models," *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1087–1093, July 2001.
- [8] K. Rohr, H. S. Stiehl, R. Sprengel, J. Buzug, T. M. and Weese, and M. H. Kuhn, "Landmark-based elastic registration using approximating thin-plate splines," *tmi*, vol. 20, no. 6, pp. 526–543, July 2001.
- [9] J. P. Thirion, "Fast non-rigid matching of 3d medical images," Tech. Rep., INRIA Sophia-Antipolis, 1995.
- [10] M. Bro-Nielsen and C. Gramkov, "Fast fluid registration of medical images," in *Visualisation in biomedical computing (VBC'96)*, Hamburg, Sep. 1996, vol. 1131 of *Lect. Notes Comp. Sci.*, pp. 267–276, Springer-Verlag.
- [11] M. Holden, D. L. G. Hill, E. R. E. Denton, J. M. Jariosz, T. C. S. Cox, and D. J. Hawkes, "Voxel similarity measures for 3D serial MR brain image registration," in *Information Processing in medical imaging*, A. Šámal Kuba and A. M. Todd-Pokropek, Eds., vol. 1613 of *Lect. Notes Comp. Sci.*, pp. 472–477. Springer-Verlag, Berlin · Heidelberg, 1999.
- [12] A. Roche, G. Malandain, X. Pennec, and N. Ayache, "Multimodal image registration by maximization of the correlation ratio," Tech. Rep., INRIA Sophia-Antipolis, 1998.
- [13] C. Studholme, D. L. G. Hill, and Hawkes D. J., "Multi-resolution voxel similarity measures for MR-PET registration," in *Proc. IPMI'95*, pp. 187–198. Kluwer Academic Publishers, Netherlands, 1995.
- [14] C. Studholme, Hawkes D. J., and Hill D. L. G., "An overlap invariant entropy measure of 3d medical image alignment," *Pattern Recognition*, vol. 32, no. 1, pp. 71–86, 1999.
- [15] A. Roche, G. Malandain, N. Ayache, and S. Prima, "Towards a better comprehension of similarity measures used in medical image registration," in *Proc. MICCAI'99*, Berlin · Heidelberg, 1999, vol. 1679 of *Lect. Notes Comp. Sci.*, pp. 555–566, Springer-Verlag.
- [16] L. A. Segel, *Mathematics applied to continuum mechanics*, Dover classics of Science and Mathematics. Dover Publ., New York, 1987.
- [17] G. E. Christensen, Rabbitt R. D., and M. I. Miller, "Deformable templates using large deformation kinematics," *IEEE Trans. Medical Imag.*, vol. 5, no. 10, pp. 1435–1447, Oct. 1996.
- [18] P. G. Ciarlet, *Mathematical Elasticity Volume I: Three-Dimensional Elasticity*, North-Holland, amsterdam edition, 1993.
- [19] G. Wollny, "Mia - a toolchain for medical image analysis," <http://mia.sourceforge.net>, 2002.
- [20] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, Cambridge University Press, New York, second edition, 1992.
- [21] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. M. Donato, J. Dongarra, V. Eijkhout, R. Pozo, Ch. Romine, and H. Van der Vorst, "Templates for the solution of linear systems: Building blocks from iterative methods," SIAM, <http://www.netlib.org/templates/templates.ps>, 1993.
- [22] G. Engeln-Müllges and F. Reutter, *Formelsammlung zur Numerischen Mathematik mit Turbo Pascal Programmen*, Bibliographisches Institut & F. A. Brockhaus AG, Mannheim, 3. aufl. edition, 1991.
- [23] C. C. Paige and M. A. Saunders, "Solution of sparse indefinite systems of linear equations," *SIAM Journal of Numerical Analysis*, vol. 12, no. 4, pp. 866–869, 1975.
- [24] R. Southwell, *Relaxation Methods in Engineering Science - a Treatise in Approximate Computation*, Oxford University Press, 1940.
- [25] M. Bro-Nielsen, *Medical Image Registration And Surgery Simulation*, Ph.D. thesis, Technical University of Denmark, 1996.
- [26] B. Greer and G. Henry, "High performance software on intel pentium pro processors or micro-ops to teraflops," <http://www.supercomp.org/sc97/program/TECH/GREER/INDEX.HTM>, 1997.

TABLE II  
RATIO OF COST OF ITERATIVE METHODS W.R.T. CONV

	CONV <sub>3</sub>	CONV <sub>5</sub>	CONV <sub>7</sub>	CONV <sub>9</sub>
SOR	8	35	94	199
MINRES	5	20	53	113

Number of iterations at which the computational cost of the iterative methods exceeds the cost of convolution filters. If the number of iterations to solve (7) can be restricted, using CONV with high filter order does not offer an advantage over the iterative methods.

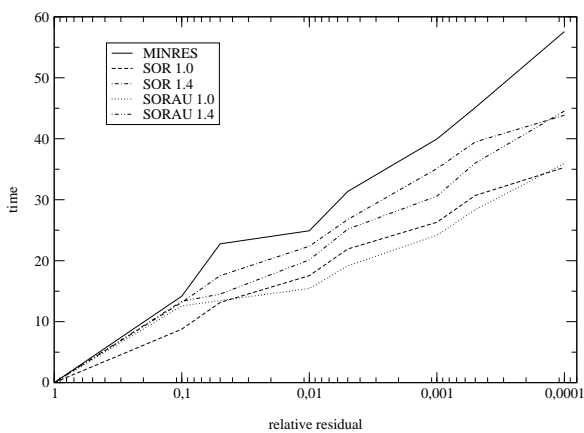
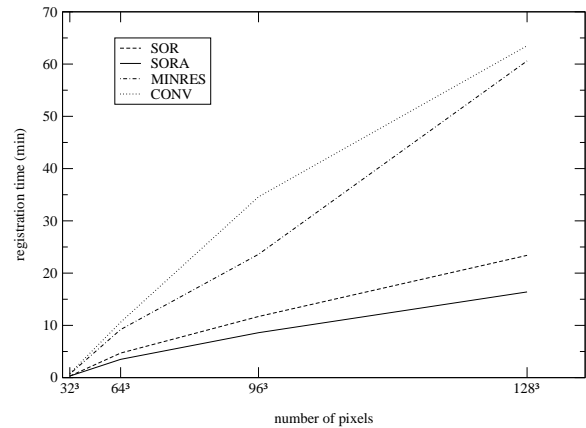
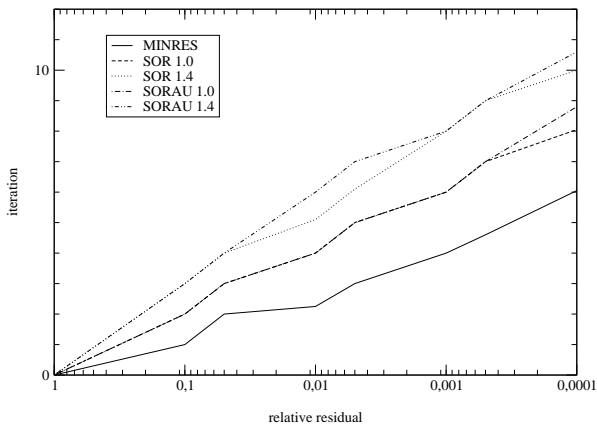


Fig. 3. Scaling of the execution times with image size: All methods show a sub-proportional time increase w.r.t. images of size  $64 \times 64 \times 64$ . SORA scales best.

Fig. 1. MINRES shows the best convergence properties in term of iteration count (top). However, in terms of execution time, SOR and SORA perform best (below), because their cost per iteration is well below that of MINRES.

TABLE I  
COMPUTATIONAL COST TO SOLVE (7)

$\frac{\text{SOR}}{\text{iteration}}$	$\frac{\text{SORAU}}{\text{iteration}}$	$\frac{\text{MINRES}}{\text{iteration}}$	CONV <sub>3</sub>	CONV <sub>5</sub>	CONV <sub>7</sub>	CONV <sub>9</sub>
$O(66n)$	$\leq O(66n)$	$O(117n)$	$O(486n)$	$O(2250n)$	$O(6174n)$	$O(13122n)$

The individual cost of iterative methods is given per iteration. Subscripts of CONV denote the filter width. Note the cost of MINRES is nearly doubled w.r.t. SOR, and the cost of CONV increases rapidly with larger filter widths.

TABLE III  
TIME AND ACCURACY FOR SOLUTION OF PDE (7) USING CONV

	3		5		7		9	
time (s)	23.9	(0.1)	111.4	(0.3)	312.5	(0.5)	723.4	(1.0)
$\frac{\text{res}(v_{\text{sol}})}{\text{res}(v=0)}$	0.44	(0.01)	0.42	(0.04)	0.40	(0.07)	0.39	(0.13)

Average registration time (variance) and average relative residuum (variance) respectively, for solution of PDE (7) for 20 randomly generated  $128 \times 128 \times 128$  force fields using convolution filters (CONV) w.r.t. filter width. Note that CONV is not able to solve (7) accurately using small filter widths.

TABLE IV  
REGISTRATION OF SYNTHETIC IMAGES (ITERATIVE METHODS)

$\epsilon$ max iterations	MINRES		SOR		SORA							
	0.01	0.005	0.01	0.005	0.01	0.005						
10	9.2	(1.2)	10.2	(2.0)	4.7	(1.0)	4.7	(1.3)	<b>3.5</b>	(0.7)	<b>3.5</b>	(0.7)
	10.4	(1.2)	10.2	(1.4)	10.6	(1.3)	10.5	(1.3)	10.3	(1.3)	10.3	(1.3)
20	13.4	(2.8)	19.8	(6.3)	8.0	(1.6)	9.3	(1.8)	<b>5.7</b>	(1.0)	<b>5.8</b>	(1.0)
	10.6	(0.9)	10.9	(1.1)	10.2	(1.1)	10.4	(1.1)	10.0	(0.9)	10.2	(1.1)
80	<b>26.0</b>	(4.5)	<b>31.5</b>	(15.3)	53.5	(17.4)	55.3	(16.9)	51.8	(10.7)	44.2	(10.3)
	10.5	(0.9)	9.1	(2.5)	10.8	(0.8)	10.6	(0.9)	11.0	(0.7)	10.7	(0.7)

A registration of synthetic 3D-images of size  $64 \times 64 \times 64$  was performed. The respective registration time (upper row in cell), and the registration accuracy  $A_{\text{reg}}$  (lower row) were averaged over 20 registration experiments (variance in parenthesis). The best registration times per maximal number of iterations are emphasized using bold fonts. Registration accuracy using only 10 iterations to solve (7) is nearly as good as using more iterations. With a limited number of iterations, SORA performs best.

TABLE V  
REGISTRATION OF SYNTHETIC IMAGES (CONVOLUTION FILTERS)

	CONV <sub>3</sub>	CONV <sub>5</sub>	CONV <sub>7</sub>	CONV <sub>9</sub>				
time (min)	4.4	(0.5)	10.6	(2.5)	28.5	(8.0)	60.3	(13.7)
$A_{\text{reg}}$	5.0	(1.7)	10.9	(1.9)	11.2	(1.5)	11.3	(1.1)

Registration of synthetic images of size  $64 \times 64 \times 64$  using convolution filters was performed. The respective registration times (upper row in cell) and registration accuracy (lower row) were averaged over 20 registration experiments. Note the low accuracy obtained for CONV<sub>3</sub>.



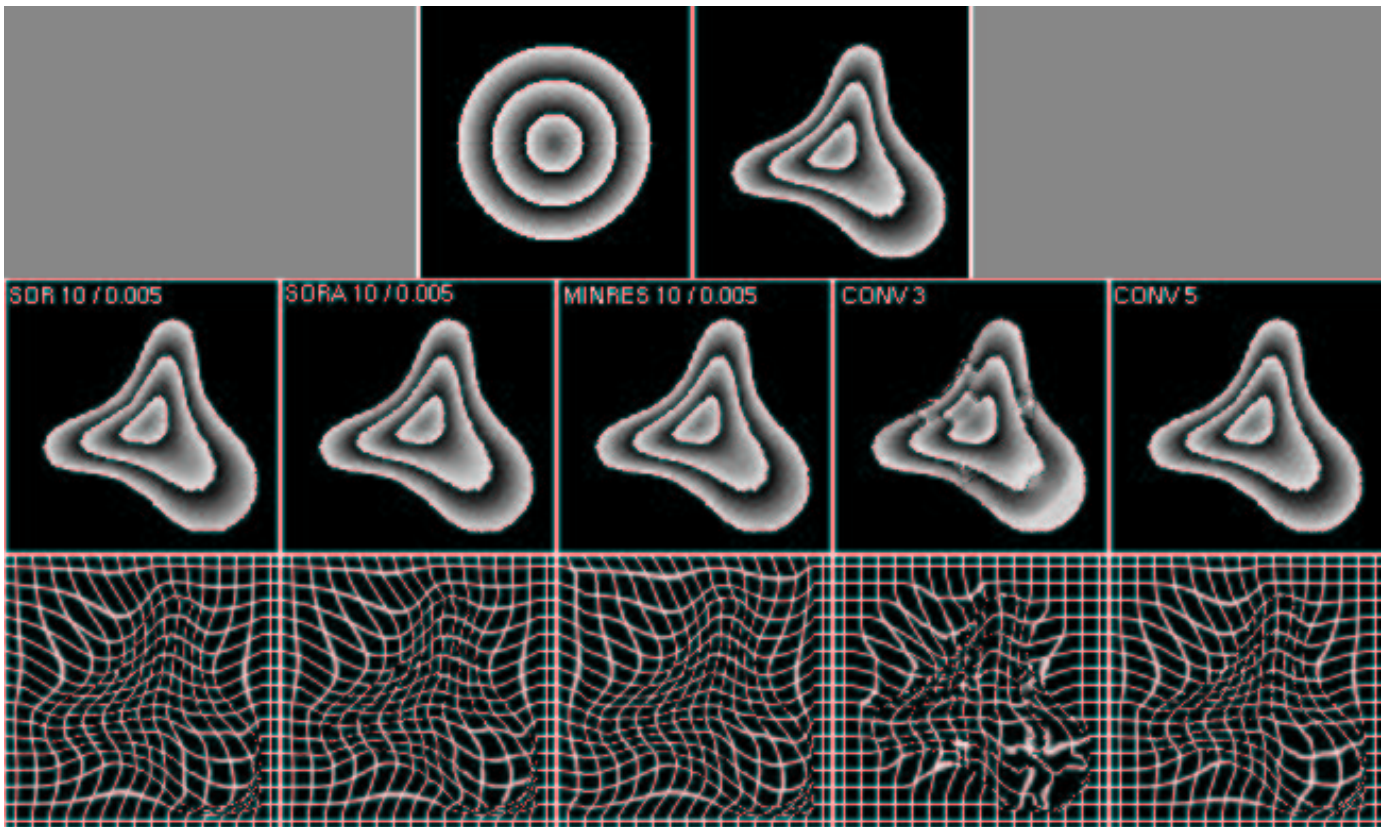


Fig. 2. Registration of a synthetic study (top-left) and an example reference (top-right). Results are shown in the middle row, and the obtained deformations in the lower row. Iterative methods were applied using a limit of 10 iterations and  $\varepsilon = 0.01$ . Note the registration error when using a small filter width (CONV<sub>3</sub>).