A Parallel Implementation of Non-Rigid Registration Using a Volumetric Biomechanical Model

M. Sermesant[1], O. Clatz[1], Z. Li[2], S. Lantéri[2], H. Delingette[1], N. Ayache[1]

[1] Epidaure Research Project
[2] Caiman Research Project
INRIA Sophia-Antipolis, 2004 route des Lucioles
06902 Sophia-Antipolis, France

Abstract. Because of the accuracy required in a neurosurgical procedure, tracking intra-operative deformations is a challenging task. Furthermore, the clinical demand for fast non rigid registration will have to be met in a very near future. In this paper, we propose a patient-specific biomechanical model based on block-matching in order to register two MR images of the same patient with a parallel implementation. Compared to other intra-operative registration techniques, this method combines a viscoelastic mechanical regularization with a correlation-based iconic energy term. We first shortly present the theoretical aspects of our method (more detailed in [9]). Then we describe in more details the parallel implementation of the algorithm. Finally we present a retrospective registration study made of four pre/post operative MRI pairs.

1 Introduction

Non-rigid registration aims at deforming one image onto another image. However, even if the registered image has the same grey levels as the target image, the computed deformation may be very unrealistic. Capturing a realistic deformation field is important to follow critical structures.

In this paper, we introduce a biomechanical model in a registration framework, thus enabling to add anatomical and mechanical *a priori* knowledge. In this way, we indirectly constrain the displacement field to follow a realistic range of deformation.

Because biomechanical deformation computation can also be performed efficiently, it has been used to analyze per-operative brain deformations [2, 10, 7]. We propose a novel registration method which dynamically combines images and mechanical information. Our method relies on a dynamic law of motion of a tetrahedral mesh, which leads to iteratively solving a linear system of equations.

To have a computation time suitable for clinical use, we chose to rely on parallel computing since it has proved to be both powerful [11] and economically viable.

2 Non-Rigid Registration Method

The novelty in our registration algorithm is to integrate anatomical and mechanical *a priori* knowledge through a volumetric biomechanical model of the brain. The image information is used as force boundary conditions and external loads applied to the model. The aim of these forces is to deform the biomechanical model in such a way that the induced displacement field brings the reference image towards the target image.

In a deformable model framework, there is a balance between mechanical and iconic influences during the full registration process through an energy minimization.

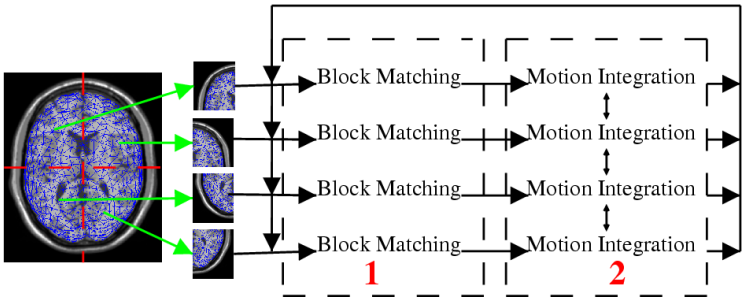

Fig. 1. Parallel implementation of the registration algorithm, distributing image backs and parallelizing the computation of image forces and the solution of a linear system of equations.

2.1 Internal Energy: Biomechanical Model

The internal energy regularizes the displacement field and is based on continuum mechanics, more precisely on linear elasticity. This simple model guarantees that an area will follow the motion of its neighbors without image matching computed in this area.

2.2 External Energy: Block-Matching Displacements

The external energy is based on intensity correlation such that each vertex is moved by an image force towards a voxel of similar intensity in the target image. Until now, deformable models mainly used gradient or distance map-based forces applied on surface vertices along the normal. We propose to use a volumetric external energy based on 3D block matching for two reasons :

- full motion of the organ can be registered, not limited to apparent motion, as with forces along the surface normal.
- more complex deformation can be estimated inside the model, not limited to interpolation of the surface deformation.

The quality of the registration with such a volumetric external energy does not rely entirely on the surface segmentation accuracy, unlike most surface-based methods.

The external energy is based on the assumption that the intensity around a given vertex of the mesh should be similar in the two images, the difference only coming from noise and anatomical changes. Thus we attach to each vertex a surrounding block of voxels from the reference image and search in the target image for the most similar block around the current position. The matching voxel selected is the center of the block maximizing the chosen similarity criterion. Then, an external force computed for each vertex is proportional to the distance to the matching voxel, and directed from the vertex current position towards the position of that voxel.

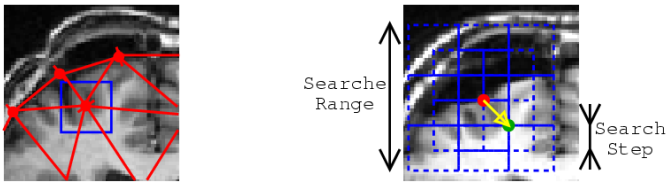

Fig. 2. (Left) Reference Image: 3D blocks of voxels initialized for each vertex in the reference image. (Right) Target Image: surrounding blocks searched for similarity measure maximum.

Several similarity measures can be used (sum of square differences, mutual information,...) depending on the type of noise and image modalities used. Moreover, this type of external energy is well-suited for a coarse-to-fine approach, by varying the search step and the search range in the target image [5].

3 Motion Integration

It is important to take into account the inertia effect, and therefore to use a dynamic equation of motion. Indeed, even if the update of the displacement field is small between two time-steps, it is much more efficient and stable to find a balance between external and internal energies with a dynamic approach.

Therefore, we integrate the motion in the dynamic equation:

$$M\frac{d^2U}{dt^2} + C\frac{dU}{dt} + KU = F \tag{1}$$

with M, C, K, the mass density ($= 1$ for the brain), damping, stiffness matrices (taken from [1]) and U the displacement from rest position ($P(t) = P(0) + U(t)$, with $P(t)$ the position at time t). F is the external force vector computed from the image matching.

3.1 Numerical Integration

We use the finite element method to numerically solve this system. Spatially, we use linear tetrahedral elements and temporally, the semi-implicit Houbolt

method. With this method, speed and acceleration are discretized as follow:

$$\frac{d^2P(t+\Delta t)}{dt^2} = \frac{1}{\Delta t^2}[2P(t+\Delta t) - 5P(t) + 4P(t-\Delta t) - P(t-2\Delta t)]$$

$$\frac{dP(t+\Delta t)}{dt} = \frac{1}{6\Delta t}[11P(t+\Delta t) - 18P(t) + 9P(t-\Delta t) - 2P(t-2\Delta t)]$$

Replacing this in eq. (1) gives a linear system to solve in $U(t+\Delta t)$ at each iteration. Since we are looking for small deformations (<10%) and small displacements [3], the stiffness matrix is chosen constant (linear elasticity).

The registration process can be decomposed in two steps:

1. the initialization procedure:
   - mass M, damping C and stiffness K matrices assembly,
   - $U(0)$, $\dot{U}(0)$, $\ddot{U}(0)$ initialization,
   - time step choice,
   - several constants computation:

$$a_0 = \frac{2}{\Delta t^2}\,;\, a_1 = \frac{11}{6\Delta t}\,;\, a_2 = \frac{5}{\Delta t^2}\,;\, a_3 = \frac{3}{\Delta t}\,;$$
$$a_4 = -2a_0\,;\, a_5 = \frac{-a_3}{2}\,;\, a_6 = \frac{a_0}{2}\,;\, a_7 = \frac{a_3}{9}$$

   - $U(\Delta t)$ and $U(2\Delta t)$ initialization with $U(0)$,
   - effective stiffness matrix computation:

$$\hat{K} = K + a_0 M + a_1 C$$

   - $\hat{K}$ preconditioner construction. Since the stiffness matrix is constant (because the time step is constant), this operation is performed only once.

   As this step should be computed pre-operatively, the corresponding execution time has not been taken into account in the results section. However, this full step takes about 3 seconds on a 1 Ghz Pentium III PC.
2. the solution of a linear system of equations at each time step:
   - effective external forces computation:

$$\hat{F}(t+\Delta t) = F(t+\Delta t) + M\,(a_2 U(t) + a_4 U(t-\Delta t) + a_6 U(t-2\Delta t))$$
$$+ C(a_3 U(t) + a_5 U(t-\Delta t) + a_7 U(t-2\Delta t))$$

   - iterative linear system solution:

$$\hat{K}U(t+\Delta t) = \hat{F}(t+\Delta t) \tag{2}$$

   - nodes positions update.

4 Implementation

4.1 Mesh Creation

The different steps of the automatic process used to create a tetrahedral mesh of the brain from a MRI are:

1. rigid registration to align the mid-sagital plane of the MRI with a coordinate plane, if the image is not already registered along a coordinate, using the algorithm proposed in [6],
2. cortex binary mask creation, using the method proposed by Mangin [4] and implemented in Brainvisa[1]. This software, downloadable on the web, is fully automatic and very efficient,
3. creation of a brain triangle mesh, from the mask isosurface extraction. The number of triangles chosen for this surface is directly related to the number of tetrahedra in the final mesh
4. creation of a brain tetrahedral mesh, using the software GHS3D[2], developed at INRIA by the Gamma Research Project.

This fully automated mesh creation step takes less than one minute (for a mesh with ≈ 40000 tetrahedra) on a 1 Ghz Pentium III PC.

4.2 Mesh Partition

The parallelization of the registration process relies on a distributed memory model. Each processor only computes the biomechanical and image forces for a subpart of the mesh. The tetrahedral mesh is partitioned using the METIS[3] software, a state-of-the-art graph partitioning tool that also includes fill-reducing reordering capabilities. At the end of the partitioning step, each vertex of the mesh has been assigned to a processor in such a way to minimize the communication cost.

4.3 Matrix Assembly

The PETSc[4] library is used for parallel matrix and vector storage, preconditioning and iterative system solving. The stiffness matrix is computed, preconditioned and distributed at the beginning of the process. For message passing operations, PETSc relies on the widely adopted Message Passing Interface (MPI)[5]. Moreover, PETSc enables to work only with global mesh vertex indices even though, internally, local indices are used for accessing the distributed data structures.

[1] http://www.brainvisa.info/
[2] http://www-rocq.inria.fr/gamma/ghs3d/ghs.html
[3] http://www-users.cs.umn.edu/~karypis/metis/
[4] http://www-unix.mcs.anl.gov/petsc/petsc-2/
[5] http://www.erc.msstate.edu/misc/mpi/index.html

4.4 Linear System Preconditioning

Different preconditioning methods are available in PETSc. Because image-based forces produce a volumetric field that can be very discontinuous, the force vector can be quite different form typical physics-based forces. Therefore, the most efficient preconditioners can be different from those used in classical linear elasticity problems. Moreover, even if the standard incomplete Cholesky factorization method probably offers the best compromise in terms of computational and memory costs, a more efficient and more costly preconditioner can be used here since the preconditioning step is performed only once for the overall registration process.

4.5 Linear System Solving

A 3D problem in not tractable for a direct system solver: the inverse of a sparse matrix is not sparse, and the required computational and memory costs to obtain and store the inverse can be prohibitive. In this context, preconditioned iterative solvers are very efficient to find an approximation of the solution.

Many iterative methods are available. Even if the stiffness matrix is symmetric positive definite, it appeared that solvers non-specific to such matrices can be more efficient as illustrated in table 1 below.

Each processor computes the external forces for the vertices corresponding to the associated submesh. Then, the distributed right hand side of system (2) is computed, and the system is iteratively solved.

Solver	*Preconditioner (time)*	*Iterations Number*	*Total solution Time*
CG	IC (0.04 s)	19	0.414 s
GMRES	ILUT (1.63 s)	11	0.387 s
BICGSTAB	ILUT (1.63 s)	7	0.365 s

Table 1. Preconditioners and solvers comparison for a mesh of 2000 nodes on a PC equipped with an Intel Pentium IV/2 GHz processor. CG: Conjugate Gradient, GMRES: Generalized Minimum RESidual, BICGSTAB: BIConjugate Gradient STABilized, IC: Incomplete Cholesky, ILUT: Incomplete Lower Upper with Threshold.

5 Results

5.1 Performance evaluation

We propose to evaluate the performance of the proposed algorithm on synthesized images. As proposed in [8], we use the finite element method to generate a gravity-induced deformation on a virtual brain from BrainWeb[6]. The average displacement of the induced deformation is 11,6mm and the maximum is 27mm.

This can be considered as the largest possible deformation during a real neurosurgical procedure (see figure 3). We finally add noise to both deformed and non-deformed images and test our algorithm on this final pair of images.

[6] `http://www.bic.mni.mcgill.ca/brainweb/`

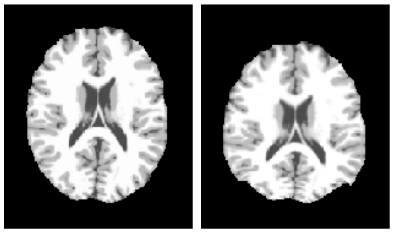

Fig. 3. C method.

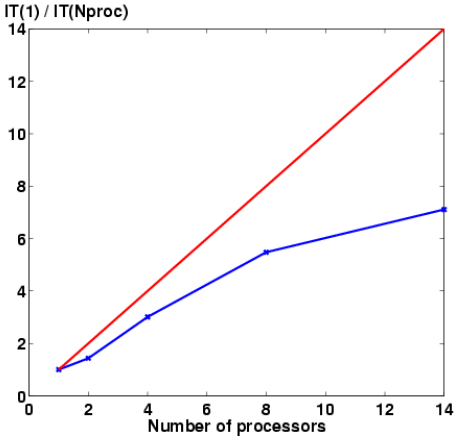

Fig. 4. Red: Ideal, Blue: f(N CPU) = Iteration Time (1 CPU) / Iteration Time (N CPU), Range of block-matching research = 5mm, research step = 2mm

Figure 4 shows the influence of the number of processors on the execution time (denoted by IT on the figure) of one outer iteration step (i.e. time step). These peformance results have been obtained on a cluster of PCs consisting of 14 Intel Pentium IV/2 Ghz processors interconnected by a Gigabit-Ethernet switch. The underlying tetrahedral mesh of the brain contains around 2000 vertices. The straight line on this figure defines the ideal parallel speedup. The rapid decrease of the parallel efficiency is simply due to the relatively coarse mesh used in our test.

Sigma	0	1	2	3	4	5	6	7
Average final error (mm)	1,31	1,33	1,33	1,33	1,35	1,33	1,43	1,41

Table 2. Influence of the noise on the robustness of the algorithm (white Gaussian noise on intensity, variable standard deviation sigma).

We then studied the influence of the damping factor on the convergence of the algorithm. We can see on figure 5.1 that the dynamic effects of the damping factor is a critical point in the registration computation process. Indeed, a small damping factor leads to an instability whereas a large damping factor leads to a large computation time. We manually estimate the right damping factor to 0,015, whereas we do not have automatic way to determine it. The optimum damping

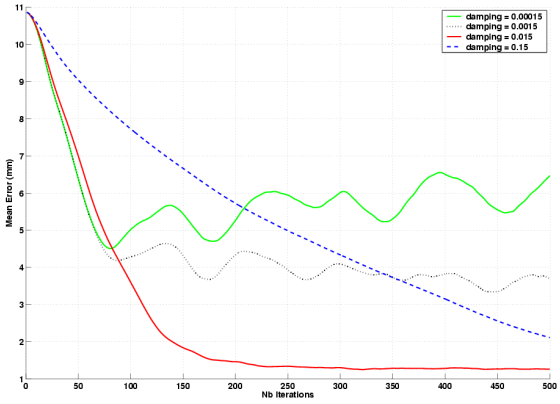

Fig. 5. Influence of the damping factor on the convergence and the mean error (in millimeters). Without enough damping, the convergence is not obtained, due to the instability. With too much damping, the convergence is importantly slowed.

factor may actually depend on the searched deformation. This damping factor authorizes an average error of 1,22mm ($10,5\%$ error) in less than 3 minutes on 14 CPU.

We present in table 2 a study of the robustness to noise of the algorithm. We can see that the algorithm, presented here with the correlation coefficient similarity measure, performs well with a high degree of noise on images. We can even consider it to be nearly not affected by noise in the common range of noise on an MRI ($3 < \sigma < 5$)

5.2 Tests on Real Images

The following results have been obtained with the fully automatic registration process previously described. In addition to the brain creation stage, we removed all the image-based forces applied on the brain surface. This implies that the surface layer of tetrahedra only move with the biomechanical regularization. In that way, the registration result is independent of the segmentation quality. Each image have been computed in less than a minute (block-matching searching range = 5mm, searching step = 2mm, correlation coefficient, 30 000 tetrahedra in the mesh, 14 CPUs).

The results obtained on real images are very encouraging. The volumetric biomechanical registration seems to be well suited to recover a physical deformation. The difference image shows the good ability of the algorithm to deform internal structures. In addition, the algorithm performs well on the brain surface, even with no image criterion. This ability to recover the surface displacement with only internal constraints shows the efficiency of the volumetric external energy.

The artifacts due to the electrodes produce registration errors in the surrounding areas. However we can be removed from their influence in the block matching search. In this way we obtain a completely smooth displacement field in this area.

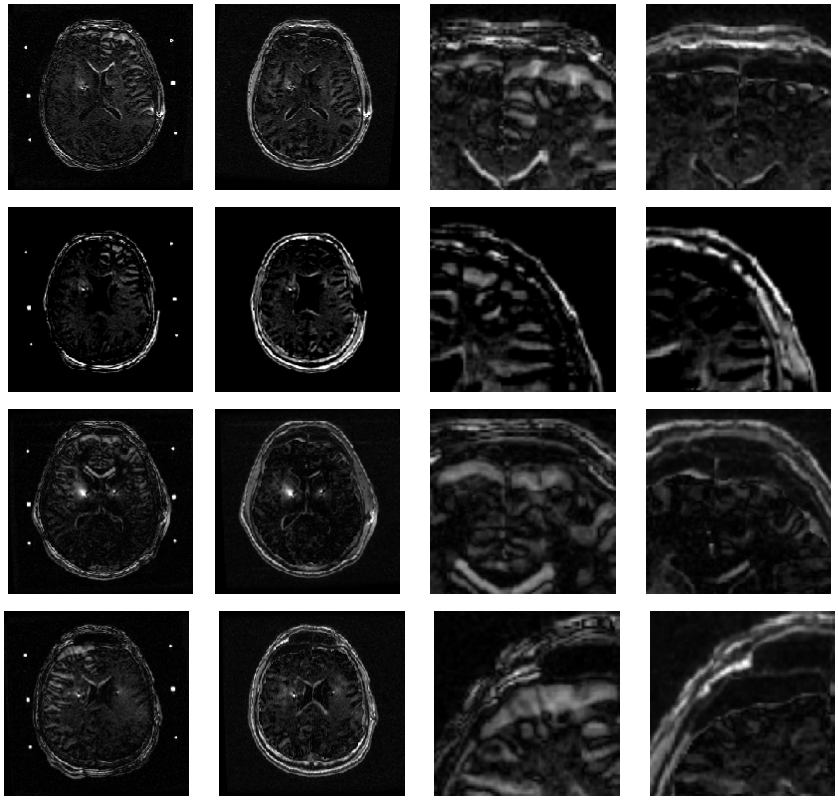

Fig. 6. Difference between pre- and post-operative images. 1st column: before registration. 2nd column: after registration. Zoom on the largest displacement area. 3rd column: before registration. 4th column: after registration.
Images come from the Neuroradiology Department, La Pitié Salpêtrière Hospital, Paris, courtesy of Pr. D. Dormont.

6 Conclusion

Using a biomechanically constrained registration method leads to a more robust method for two reasons:

- a better regularization in outliers areas thanks to the biomechanical model smoothing
- a more realistic deformation field thanks to the anatomical and mechanical knowledge included

However these advantages are computationally costly. A parallel implementation makes an intra-operative use of such methods conceivable.

Ongoing work on a dedicated PC cluster should make precise time, precision and robustness comparisons possible. Moreover, mechanical parameters, like damping, can then be extensively tested and adjusted. And on a dedicated

PC cluster, the communication time is very reduced, as the network is not slowed down by other tasks.

We also wish to pursue the work on synthesized deformations by studying the influence of the size of the mesh on the global error.

References

1. O. Clatz, H. Delingette, E. Bardinet, D.Dormont, and N. Ayache. Patient-specific biomechanical model of the brain: application to Parkinson's disease procedure. In *Surgery Simulation & Soft Tissue Modeling (IS4TM'03)*, 2003.
2. M. Ferrant, A. Nabavi, B. Macq, P. Black, F. Jolesz, R. Kikinis, and S. Warfield. Serial registration of intraoperative mr images of the brain. *Medical Image Analysis*, 6(4):337–360, 2002.
3. S. Kyriacou, A. Mohamed, K. Miller, and S. Neff. Brain mechanics for neurosurgery: Modeling issues. *Biomechanics and Modeling in Mechanobiology*, 2002.
4. J.-F. Mangin, O. Coulon, and V. Frouin. Robust brain segmentation using histogram scale-space analysis and mathematical morphology. *Lecture Notes in Computer Science*, 1496:1230–1241, 1998.
5. S. Ourselin, A. Roche, S. Prima, and N. Ayache. Block Matching: A General Framework to Improve Robustness of Rigid Registration of Medical Images. In A.M. DiGioia and S. Delp, editors, *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery (MICCAI 2000)*, volume 1935 of *Lectures Notes in Computer Science*, pages 557–566, Pittsburgh, Pennsylvanie USA, octobre 11-14 2000. Springer.
6. S. Prima, S. Ourselin, and N. Ayache. Computation of the mid-sagittal plane in 3D brain images. *IEEE Transaction on Medical Imaging*, 21(2):122–138, February 2002.
7. J. Rexilius, S. Warfield, C. Guttmann, X. Wei, R. Benson, L. Wolfson, M. Shenton, H. Handels, and R. Kikinis. A novel nonrigid registration algorithm and applications. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI'01)*, volume 2208 of *LNCS*, pages 923–931. Springer, 2001.
8. J. A. Schnabel, C. Tanner, A. Castellano-Smith, M. O. Leach, C. Hayes, A. Degenhard, R Hose, D. L. G. Hill, and D. J. Hawkes. Validation of Non-Rigid Registration using Finite Element Methods. In *In Proc. Information Processing in Medical Imaging (IPMI'01)*, volume 2082 of *Lectures Notes in Computer Science*, pages 344–357, University of California at Davis, 18-22 June 2001. Springer.
9. M. Sermesant, O. Clatz, H. Delingette, and N. Ayache. Non-rigid registration combining volumetric biomechanical model and block-matching. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI'03)*, Lecture Notes in Computer Science (LNCS). Springer, 2003. submitted.
10. O. Skrinjar, A. Nabavi, and J. Duncan. Model-driven brain shift compensation. *Medical Image Analysis*, 6(4):361–374, 2002.
11. S. Warfield, F. Talos, A. Tei, A. Bharatha, A. Nabavi, M. Ferrant, and R. Kikinis P. Black, F. Jolesz. Real-time registration of volumetric brain MRI by biomechanical simulation of deformation during image guided neurosurgery. *Computing and Visualization in Science*, 5(1):3–11, 2002.