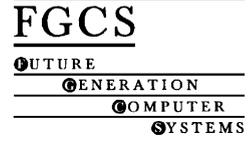




ELSEVIER

Future Generation Computer Systems 18 (2001) 115–125



www.elsevier.com/locate/future

Elastic registration of brain images on large PC-Clusters[☆]

J. Modersitzki^{a,*}, G. Lustig^b, O. Schmitt^c, W. Obelöer^d

^a *Institute of Mathematics, Medical University of Lübeck, Wallstraße 40, D-23560 Lübeck, Germany*

^b *Institute of Computer Engineering, Medical University of Lübeck, Ratzeburger Allee 160, D-23538 Lübeck, Germany*

^c *Institute of Anatomy, Medical University of Lübeck, Ratzeburger Allee 160, D-23538 Lübeck, Germany*

^d *Sun Microsystems GmbH, Eiffestraße 80, D-20537 Hamburg, Germany*

Abstract

The aim of the human neuroscanning project (HNSP) is to build an atlas of a human brain at cellular level. The database was obtained by a variety of image modalities and in particular histological sections of a prepared brain. As the preparation leads to linear and non-linear deformations of the tissue, reconstructing the essential information out of deformed images is a key problem within the HNSP. Our approach of correcting these deformations is based on an elastic matching of the images. Therefore, a parallel implementation was used, since the problem in general is computationally expensive and for very large scale digital images a huge amount of data has to be processed. As these requirements are in the range of today's grand challenges, a large PC-Cluster was used to provide the performance demands. The measurements and results presented here were obtained on a cluster of 48 Dual SMP platforms connected via a Myrinet network. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Elastic matching; Image registration; Human brain; Cluster-computing

1. Introduction

The reconstruction of deformed biological objects is a basic problem within biomedical image processing and registration. Especially, if images arise from a series of sections through a part of the human body, e.g. computer tomography (CT), magnetic resonance imaging (MRI), positron emission tomography (PET). In this paper we concentrate on particular problems of the human neuroscanning project (HNSP) at the Medical University of Lübeck. The aim of the HNSP is to produce a three-dimensional map of the human brain based on different modalities, in particular cellular information. Here, the information is derived mainly from histological sections.

The material has to be processed by different kinds of technical procedures before certain observations of biological specimens can be performed and consequently information can be analyzed. Current techniques always change the intrinsic geometry of the biological material. In the case of a whole human brain, which has to be sectioned into thousands of thin slices, laboratory processing leads to non-linear deformations. Although these deformations are small in general, they might become crucial for the reconstruction of cellular information. In order to obtain reliable spatial information these deformations have to be corrected.

Correcting images of spatially distorted objects is known as image registration. Two different approaches are common. One approach is based on the idea of representing the unknown distortion in terms of the coefficients of a fixed basis, such as piecewise linear

[☆] Expanded version of a former work presented at the HPCN'99.

* Corresponding author.

functions or higher order splines. Typically, these coefficients are determined by a least squares condition for some user prescribed landmarks (see, e.g. [4]). This approach is known as warping [17]. The second approach is based on the formulation of the problem via a non-linear partial differential equation (PDE). To solve these equations no further information on the underlying images, e.g. landmarks, is needed (see, e.g. [1,16]). We describe a method for correcting these kind of deformations based on a linear elasticity model and leading to a non-linear PDE. This so-called elastic matching method is also used in other projects, e.g. [1,2,5,6,15].

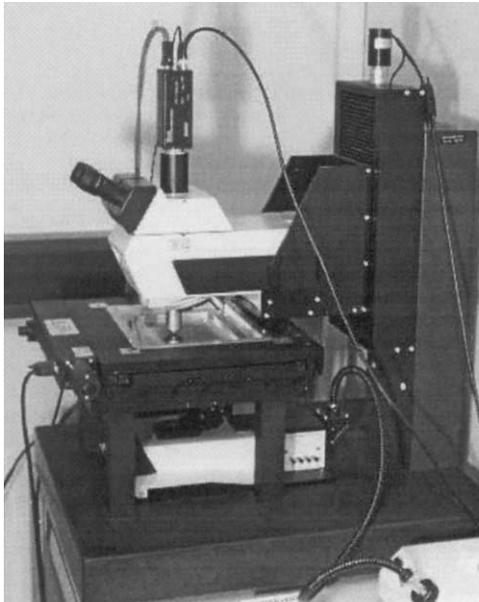
To resolve details of the brain (e.g. neurons) very high resolution scans of histological sections are produced. In comparison to MRI where the resolution is $640 \times 10^3 \mu\text{m}^3/\text{voxel}$, pictures of the HNSP offer a much higher resolution with $0.026 \mu\text{m}^3/\text{voxel}$. Matching thousands of such images in a reasonable amount of time requires computing performance that is beyond today's single processor systems. In

addition, memory requirements for the large images lead to a parallel implementation of the elastic matching algorithm. Here, with respect to an attractive price/performance ratio, we use a PC-Cluster to provide the performance demands.

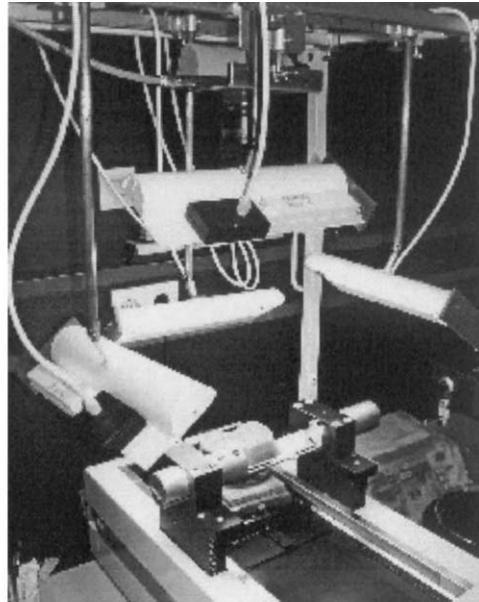
In this paper we describe the parallel implementation of our elastic matching algorithm and present performance measurements on a 48 node cluster system. Details of the HNSP are described in Section 2. After the background of the elasticity model given in Section 3, Section 4 presents internals of our parallel algorithm and performance measurements. The results and their discussion are given in Section 5.

2. The HNSP

The aim of the HNSP is the three-dimensional reconstruction of all cells of a human brain and their structural and functional characterization by means of molecular-biological techniques. The processed tissue



(a)



(b)

Fig. 1. Peripheral equipment for gaining human brain images: (a) the very large microscope (VLM) is analyzing the histological sections of the brain at the cellular level; (b) the episcope imaging device consists of a digital camera which produces images of the upper surface of the paraffin block before each sectioning process.

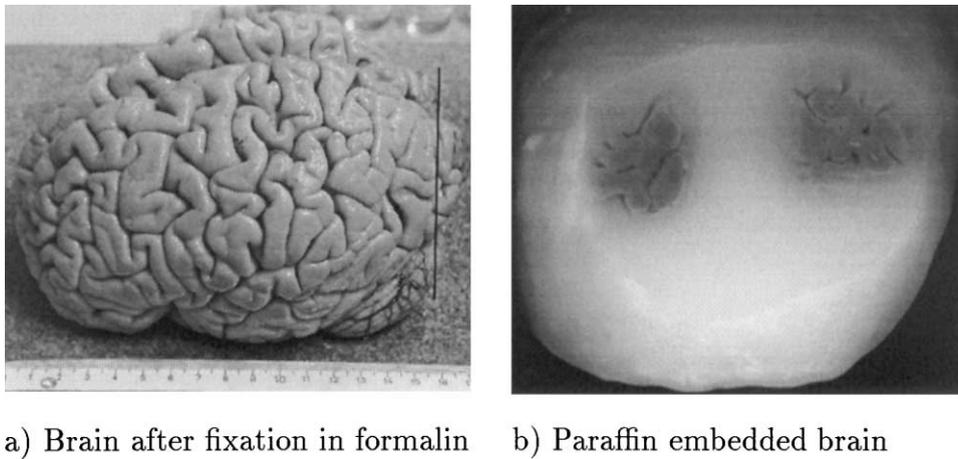


Fig. 2. The black line on the formaldehyde fixated brain (a) indicates the intersection area of the paraffin embedded brain (b) and the location of the matched sections 116–119.

is quantified by different digitization devices like high resolution transparent flat bed scanners, light microscopes (Fig. 1a) and episcopic imaging (Fig. 1b).

The data resulting from the cellular analysis should be used as the functional entity model (FEM) for the integration of the other modalities. Because data from the functional modalities have to be analyzed in different statistically independent specimens, stochastic maps of their spatial distributions have to be computed and related to the FEM. The resulting multi-modal structure function model (SFM) is of great importance for simulation studies in computational neuroanatomy because each cell in such a virtual brain is characterized by an extensive feature vector extracted from the real histological object.

In this project a 65-year-old male human post-mortem brain (Fig. 2a) of a voluntary donor was fixed in a neutral buffered formaldehyde solution for 4 months. An MRI-scan of this brain was produced after fixation. Dehydration and embedding of the brain in paraffin wax (Fig. 2b) lasted three further months [14] followed by sectioning the brain into 6214 slices with a thickness of 20 μm . Before each sectioning process a high resolution episcopic image (1352×1795 pixels, 24 bit, 7×10^6 bytes) of the section plane was scanned.

After sectioning, the tissue slices were stretched in warm water at 55°C in order to get flat tissue sections. However, these methodological steps produce

non-linear deformations of each section. The sections were stained in gallocyanine chrome alum and mounted under cover-glass for bright field light microscopy [14]. As larger sections cannot be investigated under conventional microscopes a special microscope was used. This very large and high precision microscope (VLM, Fig. 1a) was equipped with three motors and enables the exploration of the tissue volume. The VLM is controlled by a single PC which is equipped with a frame grabber in addition. This system detects automatically 10–15 optical sections with cellular objects within the section space.

At a resolution of 0.016 $\mu\text{m}/\text{pixel}$ a huge amount of data has to be processed. Therefore, a small five node cluster consisting of 400 MHz PII platforms perform the analysis (segmentation, object recognition, object morphometry, object densitometry) of the image stacks which are distributed by the VLM control PC.

The whole system (VLM+controller+cluster) is an integrated solution for a very large scale problem in computational anatomy and it is called light microscopical section analyzer (LMSA). With this system different neuronal entities can be analyzed on different scales, i.e. from macroscopic details down to the cellular level.

The stained sections were digitized using a high resolution flat bed scanner. With a resolution of 800 ppcm (or 2032 ppi) at an 8 bit grey scale mode, the size of the largest image is $11\,000 \times 7000$ pixels (about 196 MB)

[13]. The uncompressed amount of flat bed scanned data is approximately 700 GB + 40 GB episodic data for one human brain. If in addition each spatial stack of images would also be stored at a magnification of 0.016 $\mu\text{m}/\text{pixel}$, an expected data mass of about 4 PB has to be handled. A reduction of these data amount can be obtained by storing only the object feature vectors (8–12 features). This results in a reduction of the amount of data to 2.5 TB.

The stack of scanned images is the basis for computing the deformations introduced by the preparation processes. In Section 3 a mathematical formulation of the underlying problem is presented.

3. Modeling non-linear deformations of two consecutive sections

The deformation of a tissue section introduced by the preparation processes can be decomposed into a linear part (including translation, rotation, scaling, shearing) and a non-linear part. The linear part results, e.g. from fixation of the tissue section on the microscopic slide. The non-linear part results from elastic properties of the tissue and is thus called elastic deformation. From Fig. 6c, where the difference between the section 116 and the linearly corrected section 117 is shown, it can be recognized that the non-linear part might also be substantial.

To compute the elastic deformation a standard approach of linear elasticity theory was used [9]. We are looking for an elastic deformation of a template image (T) that simultaneously minimizes the difference $T(x - u, y - v) - R(x, y)$ between the deformed and the reference image (R) and the deformation energy

$$E(u, v) = \int_{\Omega} \frac{1}{2} \lambda (u_x + v_y)^2 + \mu (u_x^2 + v_y^2 + \frac{1}{2} (u_y + v_x)^2) \, d(x, y).$$

Here, the so-called deformation field $(u, v) = (u(x, y), v(x, y))$ describes the elastic deformation and μ, λ are the so-called Lamé-constants, see, e.g. [9]. This approach enforces similarity of the images as well as connectivity of the tissue. Note, in this notation $T(x - u, y - v)$ might be viewed as the non-deformed version of the template.

Applying the calculus of Euler–Lagrange we find that a minimizer is characterized by the two-dimensional Navier–Lamé equations (1) (cf. e.g. [9])

$$\begin{pmatrix} f \\ g \end{pmatrix} = \begin{pmatrix} \mu(u_{xx} + u_{yy}) + (\lambda + \mu)(u_{xx} + v_{xy}) \\ \mu(v_{xx} + v_{yy}) + (\lambda + \mu)(u_{xy} + v_{yy}) \end{pmatrix} \\ =: A \begin{pmatrix} u \\ v \end{pmatrix}. \quad (1)$$

Note that $(f, g)^T$, which might be viewed as a force field, depends non-linearly on the deformation, cf. Eq. (2)

$$\begin{pmatrix} f \\ g \end{pmatrix} = \begin{pmatrix} (T(x - u, y - v) - R(x, y))T_x(x - u, y - v) \\ (T(x - u, y - v) - R(x, y))T_y(x - u, y - v) \end{pmatrix}. \quad (2)$$

An appropriate discretization of these equations finally leads to a fix-point type equation for the unknown deformation field, cf. Eq. (3)

$$A(u^{k+1}, v^{k+1})^T = (f(u^k, v^k), g(u^k, v^k))^T. \quad (3)$$

In principle, any method for solving a system of linear equations can be used to compute the solution of Eq. (3). However, a discretization with $m \times n$ points results in $N = 2mn$ unknowns (e.g. for 512×512 discretization points we end up with $N = 2^{19} = 524\,288$) and A becomes $N \times N$. For a standard LU -decomposition one needs to store $\mathcal{O}(N^2)$ real numbers and approximately $\mathcal{O}(N^3)$ floating point operations [8]. Thus, memory and computational requirements make a parallel implementation of an iterative solver for Eq. (3) unavoidable.

4. Parallel realization and measurements

4.1. Parallel implementation of the elastic matching algorithm

The implementation of our parallel algorithm for p processes can be divided into three parts (quantities used only locally are denoted with a subscript “loc”). The repetition of part two and three is called *outer loop* (the superscript “ k ” serves as an outer iteration

Table 1
Principle phases of the parallel implementation of the CG-algorithm

(a)	Computation	Matrix-vector multiplication Inner product	$q_{\text{loc}} = Ap$ $\alpha_{\text{loc}} = q_{\text{loc}}^T p_{\text{loc}}$ $\alpha = \sum \alpha_{\text{loc}}$
(b)	Communication	Build and distribute global sum	
(c)	Computation	Two SAXPY's Inner product	$x_{\text{loc}} = x_{\text{loc}} + \alpha p_{\text{loc}}$ $r_{\text{loc}} = r_{\text{loc}} - \alpha q_{\text{loc}}$ $\beta_{\text{loc}} = r_{\text{loc}}^T r_{\text{loc}}$ $\beta = \sum \beta_{\text{loc}}$
(d)	Communication	Build and distribute global sum	
(e)	Computation	Local vector operation	$p_{\text{loc}} = r_{\text{loc}} + \beta p_{\text{loc}}$
(f)	Communication	Exchange local vectors	$p_{\text{loc}} \leftrightarrow$ right and left neighbor

counter). In our current implementation there is one process per node.

1. Partition the images R and T into p stripes and distribute these stripes to p processes. Each process sets $k = 0$, $(u_{\text{loc}}^k, v_{\text{loc}}^k) = (0, 0)$, and $(f_{\text{loc}}^{k-1}, g_{\text{loc}}^{k-1}) = (0, 0)$.
2. Each process applies the deformation $(u_{\text{loc}}^k, v_{\text{loc}}^k)$ to T_{loc} , i.e. it computes $T(x - u_{\text{loc}}, y - v_{\text{loc}})$ by using bilinear interpolation schemes. In addition, the forces $(f_{\text{loc}}^k, g_{\text{loc}}^k)$ (see Eq. (2)) were computed independently. The difference between the new and the old force field might be used as a stopping criteria. Note that computing this difference implies a global communication in each iteration step.
3. Solve the system of linear equations (3) for the new deformation field (u^{k+1}, v^{k+1}) . Set $k \rightarrow k + 1$ and continue with step 2.

As already pointed out, the main computational work is needed for solving Eq. (3), which has to be done in each step. Here, a parallel implementation of the conjugate gradient method (CG) is used (cf. e.g. [8]). This iterative scheme leads to an additional so-called *inner loop*.

The basic structure of our implementation is given in Table 1. The essential computational and communication costs needed in one step of the inner CG iteration are next neighbor communication (exchange local stripes with two neighbors), two global sums (inner products $\approx 4N$ FLOPS), three local SAXPY operations ($Y = \alpha X + Y$, $6N/p$ FLOPS for vectors X, Y of length N/p), and one matrix vector multiplication. Exploiting the special structure of the matrix A , the expense of this multiplication is $\mathcal{O}(N/p)$ FLOPS but depends on the particular discretization of the deformation field (u, v) . Note, N is the total number of

unknowns (i.e. $N = 2mn$ for $m \times n$ discretization points), p is the number of processes.

To process on a pair of 1024×1024 pixel images, a computation power of approximately 10 GFLOPS is needed in one outer iteration step. The number of outer iteration steps needed is in the order of 100, but depends generally on the resolution of the images and the magnitude of the deformation. Since our goal is the matching of a whole brain consisting of about 6000 sections, a multicomputer is needed to provide the enormous amount of computation power.

4.2. Hardware environment

Clusters built from commodity off-the-shelf PCs or workstations are an attractive alternative to “real” parallel machines, due to their better price/performance ratio. This is especially true, when the nodes are connected via high speed networks which combine low latencies with a high transfer bandwidth.

Our parallel implementation was performed on the “Störtebeker Cluster” [10]. This cluster consists of 48 dual SMP platforms (333 MHz Pentium II) which are interconnected via Myrinet [3]. Basic building blocks of the network topology (Fig. 3) are four switch fabrics, forming a fully connected graph. One fabric hosts eight interconnected 8-port switches arranged as a bi-graph with two stages of four switches each.

4.3. Programming interface

While commodity off-the-shelf cluster nodes deliver increasing high computation power, achieving adequate high speed communication is still a challenge. At the network interface side, it was found earlier that the “traditional” protocols and operating system

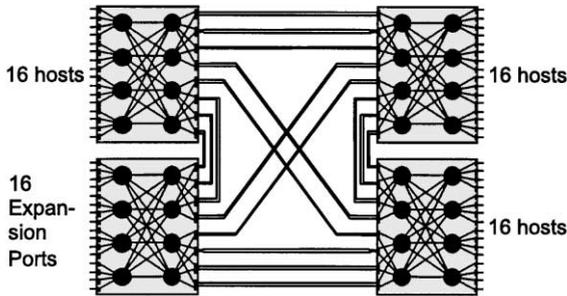


Fig. 3. Topology of the Störtebeker Cluster.

services are not able to make the very low latencies and a high bandwidth of the network technologies available at the user level. Applications often can exploit the performance of the underlying network only by using tailor-made programming interfaces. With these special interfaces and access methods, it is possible to produce and consume communication traffic close to the peak performance of system busses and network links. For most technologies, this involves direct access to the communication hardware. Unfortunately, the efficiency of communication via direct hardware

access depends strongly on programming details of the specific device technology. This dependency makes porting the applications between different technologies difficult if not impossible.

In order to provide efficient direct access without losing portability of the applications, a lean message passing programming interface has been developed at the University of Lübeck [12]. This high performance cluster communication (HPCC) environment acts as a common abstraction layer for various high speed technologies. Its portability enables applications to access different networks in a uniform and efficient manner. Currently HPCC supports direct access for four high speed networks [10]: Myrinet, Scalable Coherent Interface (SCI), Gigabit Ethernet, and Heterogeneous Interconnect (HIC).

HPCC provides the functionality for communication (connection establishment, send, receive) and dynamic process creation, similar to PVM or MPI. In order to exploit the performance of high speed devices, a zero-copy, unbuffered send, and a corresponding receive operation are offered besides the buffered, blocking send and receive operations. For the measurements shown in Fig. 4, the parallel program uses HPCC as communication layer under LINUX.

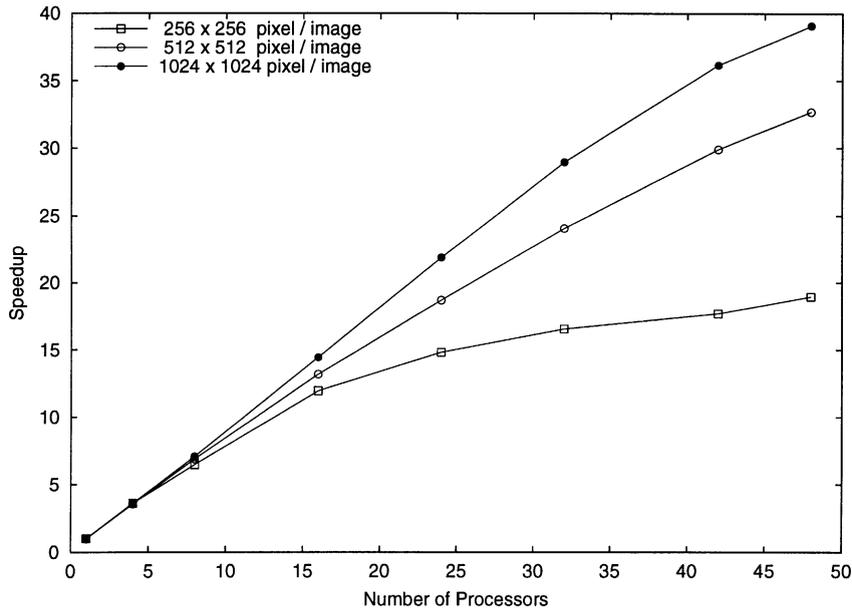


Fig. 4. Speedup for matching two pairs of $n \times n$ pixel images, $N = 256, 512,$ and 1024 .

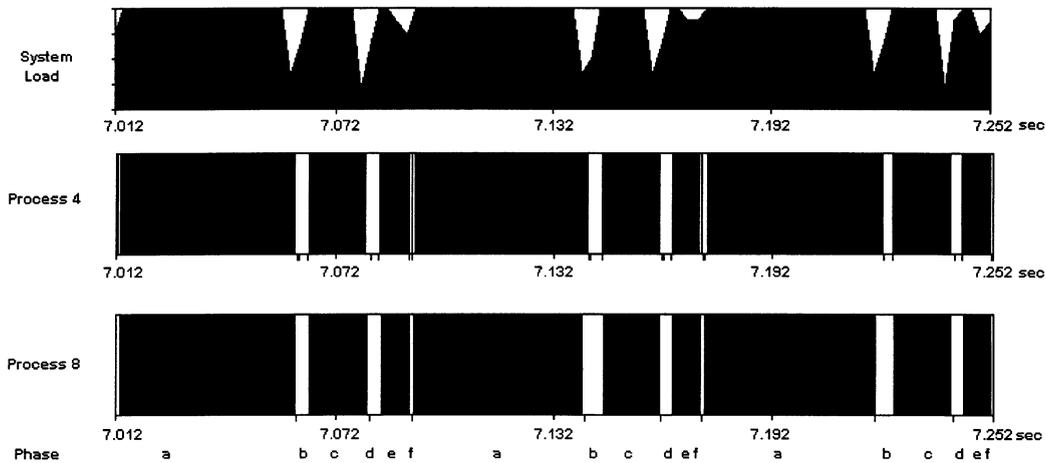


Fig. 5. System load and computation/communication phases of the arbitrarily chosen processes 4 and 8 for an eight node configuration.

4.4. Measurements

With respect to reasonable measuring times, the number of steps in the outer loop as well as the maximum number of the CG steps in the inner loop were set to 50. Measurement series with a variable number of nodes (the current implementation only supports one process per node) were performed. In order to investigate the implementation behavior at different image sizes, the speedup for three pairs (256×256 , 512×512 , and 1024×1024 pixel images) were regarded.

The run time for matching a pair of $n \times n$ pixel images with our sequential version is about 26 min for $n = 256$ and about 2 h for $n = 512$. For $n = 1024$ the time would be about 8 h. However, memory requirements prohibit this execution. In order to get the program speedup for these images irrespective of the missing normalization factor, the run time on a four node version provides the calculation base. As the speedup measurement for $n = 256$ and 512 were nearly the same on a four node system, we assume that this would be more than ever the same for $n = 1024$.

Since the parallel version also has a sequential part (i.e. starting the processes, initialization and distribution of the images, etc.) it is clear that the speedup cannot be linear (optimal). As expected, the speedup becomes better with increasing problem size (Fig. 4). For matching a pair of 1024×1024 pixel images the program speedup on 48 computation nodes is near 40 with a total run time of about 12 min.

To get insight into the program behavior, three iteration steps of the CG-algorithm are shown in Fig. 5. For observation and evaluation of the behavior the performance monitoring tool DELTA-T [11] is used. The *system load* and computation/communication phases of two arbitrarily chosen processes are shown for an eight node configuration (i.e. eight processes). Fig. 5 shows about 0.25 s of the execution time. In particular, three principle phases of the CG-algorithm as introduced in Table 1 are displayed. The upper curve in Fig. 5 shows the *system load*. Here, 100% indicates that all eight processors are working concurrently. An average utilization of nearly 100% is reached, i.e. the parallel implementation of the CG-algorithm is able to use almost the full system power. The two Gantt graphs of processes 4 and 8 show computation phases (black) and communication phases (white). Typically, more than 50% of the execution time of one CG step is taken by phase (a). Building the global sums (phases (b) and (d)) takes more time than exchanging the local vectors in phase (f), although much more data have to be transferred in phase (f).

If more computation nodes are used, the computation time of phases (a), (c), and (e) is reduced. In contrast, the time for the communication phases (b) and (d) is increased. For larger numbers of nodes this leads to a lower efficiency and might become crucial while matching two images of relatively small sizes. However, the images to be matched in the

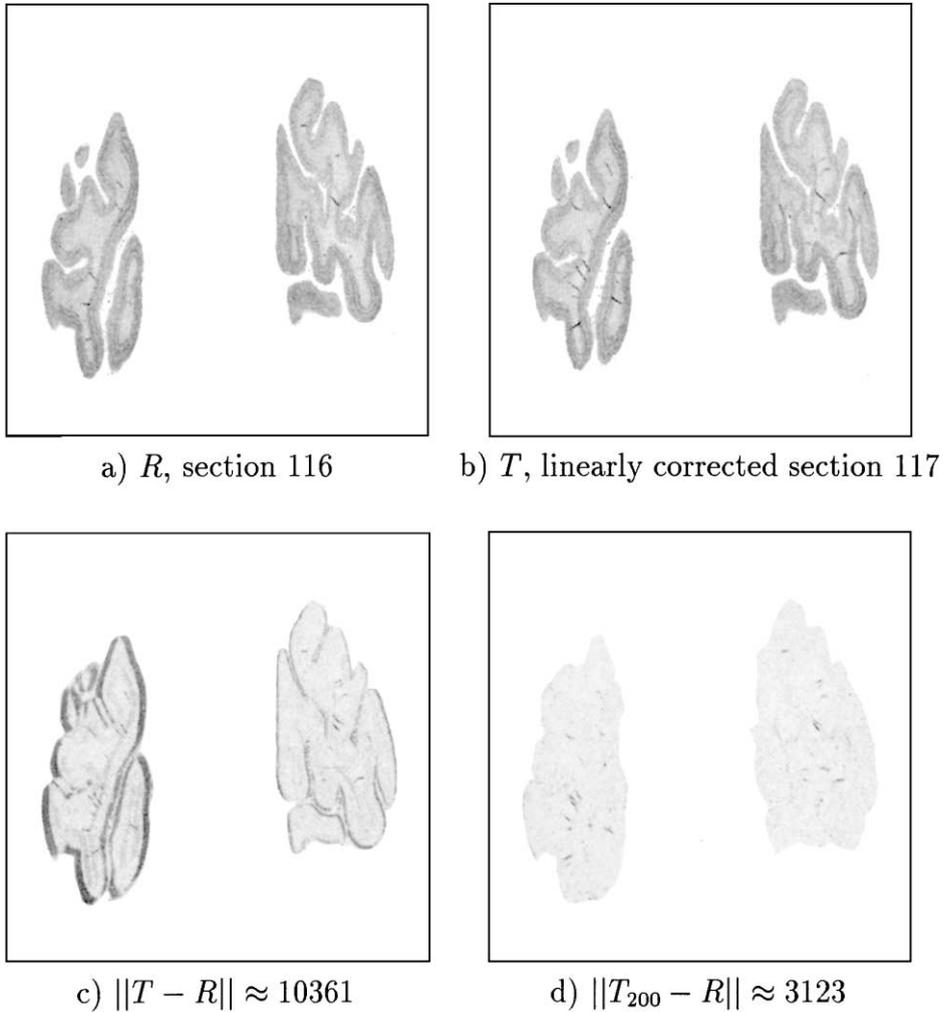


Fig. 6. (a) Reference image R (section 116) and (b) template image T (linearly corrected section 117), both 512×512 pixels, 256 gray levels. Note that differences can hardly be seen by human eyes. Difference (c) before ($\|T - R\| \approx 10361$) and (d) after ($\|T_{200} - R\| \approx 3123$) performing the elastic matching algorithm. The difference reduction ratio is $\|T_{200} - R\|/\|T - R\| \approx 30.1\%$.

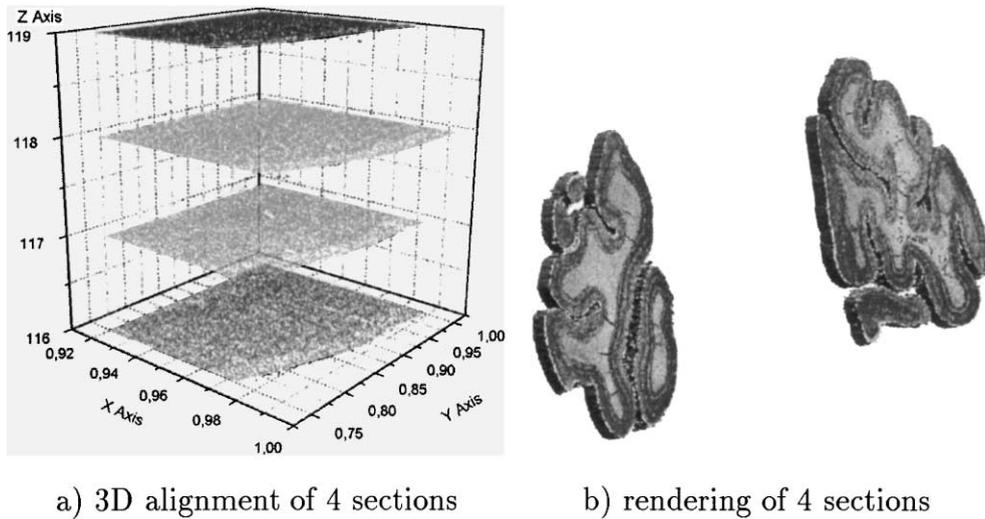
HNSP project have a high resolution and thus large sizes.

5. Results

In Fig. 6 the results of matching a pair of 512×512 pixel images are shown. Here, the arbitrarily chosen sections 116 (Fig. 6a) and 117 (Fig. 6b) out of a total of 6214 sections were matched. The sections are obtained from the occipital lobe (visual

system). Due to a better presentation and in correspondence to our measurements, these images were scaled down from 6500×2300 to 512×512 pixels.

Comparing the (linearly corrected) template 117 with the reference image 116 shows a difference of $\|T - R\| \approx 10361$ in the Frobenius norm (Fig. 6c). Using our elastic matching algorithm, we are able to reduce the difference to $\|T_{200} - R\| \approx 3123$ (Fig. 6d), which is ca. 30.1% of the initial difference. Here, we performed 200 outer iteration steps.



a) 3D alignment of 4 sections

b) rendering of 4 sections

Fig. 7. Results after elastic matching of sections 116–119: (a) three-dimensional region of interest of the object data (three-dimensional alignment of four sections); (b) three-dimensional visualization of the alignment of scanned and down-scaled 1024×1024 pixel images (scaled z-axis, rendering of four sections).

The elastic matching algorithm enables a spatial alignment of several consecutive sections. This is apparent from Fig. 7b, where four consecutive sections of 1024×1024 pixel images were matched. Note, for presentation purpose the z-axis has been scaled differently. For this example, the registration is perfect up to pixel size.

Interpolation schemes are used in order to obtain spatially corrected object data (Fig. 7a). From this, we obtain a sufficient reconstruction of the spatial geometry of a whole brain on a cellular level.

From the morphological point of view this can be considered as an adequate result. Irrespectively to the fact that there exists no other method for solving this kind of deformation problem, this technique can be considered as a sophisticated procedure to match images of tissue sections of the whole human brain.

6. Conclusion

The presented elastic matching method allows reconstruction of deformed images. Because of its universality, the technique is also useful to align images from other modalities like PET, MRI, electroencephalography (EEG). Moreover, this method

can be adapted easily to a variety of more general image registration problems.

A parallel implementation makes the approach practicable and attractive for medical image processing applications, especially for those emerging from the HNRP. Our implementation demonstrates that the elastic matching algorithm produces promising results in a reasonable amount of time on a high performance cluster system. In addition, the implementation allows matching of images obtained from high resolution section scans. Artifacts arising from down-scaling and interpolation schemes can be reduced by using large sized images.

From the medical point of view, further work has to be performed by investigating and interpreting the results. Till now we do not know what is actually the best match in medical sense, what is under-matched and which result is over-matched (corrected template and reference are identical). From the mathematical point of view a convergence proof of the overall algorithm is under work.

However, the method is a promising tool for the reconstruction process within the HNRP. In the future we will also parallelize the method using a multi-grid solver and in particular a direct solver based on fast Fourier-type techniques (FFT) [7]. Therefore, different

modules for solving the system of linear equations with appropriate parallelization strategies have to be supplied.

The presented results are based on the interaction of computer science and medicine. Sophisticated techniques are needed to produce high quality images. However, spatially non-deformed images seem to be out of reach at least in the near future. Thus, within the deformation process, specialized algorithms and its parallel realizations are needed to obtain reliable object data.

Acknowledgements

We are indebted to A. Folkers and J.-M. Frahm for implementing parts of the algorithm.

References

- [1] Y. Amit, U. Grenander, M. Piccioni, Structural image restoration through deformable templates, *J. Am. Statist. Assoc.* 86 (414) (1991) 376–387.
- [2] R. Bajcsy, S. Kovačič, Toward an individualized brain atlas elastic matching, MS-CIS-86-71 Grasp Lap 76, Department of Computer and Information Science, Moore School, University of Philadelphia, Philadelphia, PA, 1986.
- [3] N.J. Boden, D. Cohen, R.E. Felderman, A.E. Kulawik, C.L. Seitz, J.N. Seizovic, W.-K. Su, Myrinet: a gigabit-per-second local-area network, *IEEE Micro* 15 (1) (1995) 119–128.
- [4] F.L. Bookstein, A statistical method for biological shape comparisons, *J. Theoret. Biol.* 107 (1984) 475–520.
- [5] M. Bro-Nielsen, Medical image registration and surgery simulation, Ph.D. Thesis, IMM, Technical University of Denmark, Denmark, 1996.
- [6] G.E. Christensen, Deformable shape models for anatomy, Ph.D. Thesis, Sever Institute of Technology, Washington University, St. Louis, MO, 1994.
- [7] B. Fischer, J. Modersitzki, Fast inversion of matrices arising in image processing, *Numer–Algorithms* 22 (1999) 1–11.
- [8] G.H. Golub, C.F. van Loan, *Matrix Computations*, 2nd Edition, The Johns Hopkins University Press, Baltimore, 1989.
- [9] M.E. Gurtin, *An Introduction to Continuum Mechanics*, Academic Press, Orlando, 1981.
- [10] Störtebeker Cluster Project, Institute of Computer Engineering, Medical University of Lübeck, 1998, <http://www.iti.mu-luebeck.de/cluster>.
- [11] W. Obelöer, E. Maehle, Integration of debugging and performance optimization, in: *Proceedings of the Second Sino-German Workshop on Advanced Parallel Processing Technologies (APPT)*, Koblenz, 1997, pp. 17–24.
- [12] S. Petri, G. Lustig, C. Grewe, R. Hagenau, W. Obelöer, M. Boosten, Performance comparison of different high-speed networks with a uniform efficient programming interface, in: *Proceedings of the SCI Europe'99, SINTEF Electronics and Cybernetics*, Oslo (1999) 83–90.
- [13] O. Schmitt, R. Eggers, Flat-bed scanning as a tool for quantitative neuroimaging, *J. Microsc.* 196 (1999) 337–346.
- [14] O. Schmitt, R. Eggers, High contrast and homogeneous staining of paraffin sections of whole human brains for three-dimensional ultrahigh resolution image analysis, *Biotech. Histochem.* 73 (1997) 44–51.
- [15] T. Schormann, M. von Matthey, A. Dabringhaus, K. Zilles, Alignment of 3-D brain data sets originating from MR and histology, *Bioimaging* 1 (1993) 119–128.
- [16] J.-P. Thirion, Non-rigid matching using demons, *Correspondence on Computer Vision and Pattern Recognition*, IEEE Press, New York, 1996, pp. 245–251.
- [17] A.W. Toga, *Brain Warping*, Academic Press, San Diego, CA, 1999.



Jan Modersitzki is a research assistant at the Institute of Mathematics of the Medical University of Lübeck, Germany. He received his Diploma and PhD degree from the University of Hamburg in 1990 and 1995, respectively. His research interests include numerical analysis (in particular iterative methods for large linear systems), parallel computing, and digital image processing.



Gunther Lustig is a research assistant at the Medical University of Lübeck, Germany. He received his MS degree in Electrical Engineering from the University of Paderborn in 1995. Prior to joining Lübeck, he interned Daimler-Benz Systems Technology Research where he worked on portable monitoring tools for embedded systems. His research interests include parallel and distributed systems, high speed networks, and computer architecture.



Oliver Schmitt was born in Hamburg, Germany on January 10, 1964. He studied history of natural sciences, philosophy, and philology from the University of Hamburg in 1983. He received his MD degree in 1991 at the University of Lübeck, Germany. From 1986 to 1991 he investigated neuron populations in the human putamen by morphometry, first and second order stereology. From 1992 to 1993

he worked as a surgeon in the Hospital of Heide in Germany. Since 1993 he works as a research assistant at the Department of Anatomy of the University of Lübeck, Germany. In 1994 the long term research project “The Human Neuroscanning Project” was founded. His current interests include image registration, computational anatomy, distributed image analysis, mathematical morphology, 3D image analysis, and automatic detection of cells in sections through the whole human brain. He was awarded by the Louise Eylmann Foundation in 1996. He is a member of the International Society for Sterology, Anatomical Society in Germany, Neuroscience Society in Germany, the Deutsche Gesellschaft für medizinische Informatik, Biometrie und Epidemiologie e.V., and the Society of Neuroscience.



Wolfgang Obelöer was a research assistant at the Medical University of Lübeck, Germany and is now with Sun Microsystems GmbH, Germany. He received his MS degree and his PhD degree in Electrical Engineering from the University of Paderborn in 1989 and 1994, respectively. His research interests include parallel and distributed system architecture, tools for parallel systems, load balancing, and high speed networks.