

Piecewise Trilinear Deformation of Tomographic Models

SÍLVIO DE BARROS MELO

Departamento de Matemática-UFPE-Recife, PE, Brasil
silvio@dmat.ufpe.br

Abstract: In this work we introduce an iterative method that deforms brain models built from tomographic images. The deformation is used for normalization purposes: individual models are deformed to match the shape, orientation and internal morphology of a reference model. In this method the individual and the reference models are each enclosed in a cube which is subdivided to form a rectangular grid. The vertices in the individual model's grid are perturbed and the contents of each cell is then trilinearly mapped into a cube. The composite of all resulting cubes form the deformed model to be compared with the reference. The perturbations on the vertices are generated by a simulated annealing optimization technique. To maximize the performance, the models are represented in a multi-resolution fashion and the method is parallelized.

1 Introduction

A considerable part of the development of our knowledge about the human body in recent years has been supported by studies conducted in an interdisciplinary fashion, mainly with the introduction of digital computers, equipment for biological data acquisition, lasers and many more tools. These studies have helped to establish a more accurate and useful modeling of the human body and have added more efficiency to the diagnosis of a variety of diseases, and more efficiency, effectiveness and less invasiveness to their treatments. Despite the enormous progress achieved lately, we have barely started understanding the highly complex nervous system and, in particular, our brain. The studies on the human brain face great difficulties not only due to its structural complexity in both microscopic and macroscopic levels, but also due to its anatomical variability across individuals and populations, as well as its changes through different scales of time. These obstacles have induced a fragmentation in the neuroscience research worldwide to an extent that it becomes difficult to fully relate the works and results from different researchers.

An earlier effort in the direction of standardization was the development of the human brain atlas by Talairach [1]. He used two landmarks (Anterior and Posterior Commissures) and other structures to establish a three-dimensional coordinate system. Any individual brain could easily be warped into this atlas and some anatomical and structural information from the atlas could be transferred to the individual brain. Although popular and useful for anatomical normalization required for surgical procedures, the atlas is known to be limited. The Talairach atlas assumed perfect inter-hemispheric symmetry (it used only one hemisphere) and excluded the brain stem and cerebellum. In addition, the brain model was extracted

from one individual. Therefore, this atlas is more accurate for areas of the brain that have low inter-subject variability and for sites close to the landmarks of the reference system. Highly variable and asymmetrical portions of the brain in the cerebral cortex call for a more general approach.

A comprehensive, coordinated effort for providing researchers with access to the latest information is being conducted by researchers from institutions such as UCLA, Montreal Neurological Institute, UT San Antonio, Stanford, and others, under the name of Human Brain Project (HBP). Their goal is to develop a comprehensive map of the human brain that incorporates structural information (Magnetic Resonance Imaging-MRI, for instance) and functional information (Positron Emission Tomography-PET, for instance). The project is a probabilistic model that accounts for variability across individuals and groups, and the database is organized under several attributes, such as sex, age, race, etc. Any researcher will be able to arbitrate the criteria to obtain his/her desired segment of the population and have the corresponding average atlas displayed using several techniques for 3D visualization. The first step in dealing with digital images in this project and others using different individuals and modalities of acquisition, is the spatial normalization of the models to an average model. A geometric deformation is required to standardize their coordinate system and, as a second step, to minimize their variability in order to increase the power of the statistical operations. The existing techniques that perform this normalization are numerous but all suffer from different types of limitations that restrict their applicability. One of the most common limitations is their computational effort. Due to that, some of them just perform affine transformations to the entire model, while the nonlinear deformation is restricted to 2D slices that compose the

model. Others need human interference to skip difficult steps, or just wait for faster computers to appear.

2 Some Existing Techniques

Although there are many available classifications of the existing techniques, we find convenient to classify them as (1) *iterative*, (2) *landmarking* or (3) *dynamical*.

(1) In the iterative techniques, a penalty function is established between the reference model and the generated one. It drives some form of iterative optimization that affects the geometric attributes of the model, usually a deforming operator, in order to minimize the penalty function. This penalty function accounts for the geometrical differences between the models. A known reference is Collins et al. [2] in which routines were developed to alter parameters of affine maps to reduce penalty functions. Affine methods are global and do not address local morphometric variations in the brain. Iterative techniques also include the famous Free Form Deformation Sederberg et al. [3], in which the deforming model is embedded in the volume determined by a B-spline control net. When the control points are repositioned, the corresponding piecewise Bezier volume and its enclosed 3D model are deformed. This method generates smooth deformations, but the degrees of freedom for optimizing all control points may become prohibitive.

(2) In landmarking techniques, a discrete set of points related to the brain landmarks is identified in both source and target models, forming a pair wise correspondence, usually by manual interaction. At the end of the deformation, each landmark in the source model is left at the same location as its corresponding landmark in the target model. The deformation is defined by interpolation or extrapolation techniques. Methods that attempt to identify or establish automatically the landmarks (Minoshima et al. [4]) face problems with the spacial frequency of brain images. Some of the most successful methods are those that do not require a structured set of landmarks, called *scattered data deformation methods* - Ping [5]. Since a deformation is a mapping of a 3D point into another 3D point, we can also see this as a triple scalar field. Scattered deformation methods solve a triple problem of scattered data interpolation, one for each coordinate. A scattered data interpolation method solves the problem of finding a smooth function that passes through an unstructured network of points with their

respective functional values given. Any arbitrary point can be obtained as a summation of basis functions multiplied by their respective landmark. A known problem in these techniques is that, while there is a satisfactory smoothness inside the convex hull of the landmarks, the extrapolation outside is usually very poor. A remarkable method, described in Yun [6], solved that problem and was designed to be very general. In addition to the landmarks chosen by the user, the method adds deformation points on a surface surrounding the model and optimizes their positions to minimize a penalty function that accounts for smoothness outside the landmarks' convex hull. It also allows the incorporation of transfinite constraints for the user to control the shape. As of today, the method is being fine-tuned to fully work in 3D.

(3) Dynamical techniques present the most realistic modeling of the human brain. The brain model is assumed to have certain material properties and the deformation obeys physical laws such as elasticity. Forces, generated by a potential related to the models' differences, are applied to produce the deformation. A complex system of PDE's with perhaps millions of unknowns for a 3D model needs to be solved. Such complexity waits for a more feasible computational model. Some references are Miller [7] and Terzopoulos [8].

The methods are also classified as interactive (user intervention) or automatic, linear or nonlinear, non-retrospective (artificial, extrinsic image properties are used such as skin markers or various kinds of frames) or retrospective, voxel-based, or a combination of them. These properties may also be used as criteria to evaluate a method.

In this work we describe a novel approach to the anatomical standardization of brain models from a single or multiple individuals, or from multiple modalities. This iterative technique is a fully automated, parallel, nonlinear normalization of 3D models that uses piecewise trilinear deformations.

3 Description of the Standard Iterative Technique

We use simple trilinear transformations of a unit cube to achieve the deformation of our tomographic model. They map a unit cube into a volume determined by eight arbitrarily chosen points in a 3-dimensional euclidean space. Figure 1 illustrates such a volume with the vertices conveniently indexed by numbers in binary:

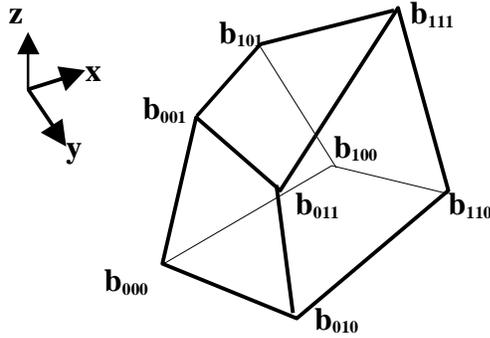


Figure 1 Trilinearly deformed cube.

The restricted trilinear transformation maps a triple (t,s,r) in the cube $[0,1]^3$ into a point $\mathbf{T}(t,s,r)$ inside the volume determined by the given points $\mathbf{b}_{000}, \mathbf{b}_{001}, \dots, \mathbf{b}_{111}$. Its computation is the application of a 3-step linear interpolation:

1. $\mathbf{b}_{jk}(t) = (1-t)\mathbf{b}_{0jk} + t\mathbf{b}_{1jk}, j,k \in \{0,1\}$
2. $\mathbf{b}_k(t,s) = (1-s)\mathbf{b}_{0k}(t) + s\mathbf{b}_{1k}(t), k \in \{0,1\}$
3. $\mathbf{T}(t,s,r) = (1-r)\mathbf{b}_0(t,s) + r\mathbf{b}_1(t,s)$

This algorithm can also be used to find the transformation's explicit form, which can be seen as the original points \mathbf{b}_{ijk} multiplied by a polynomial in the variables t,s,r of total degree 3 (the diagonal in the cube $r=s=t$ is mapped into a cubic curve, for instance.) The explicit form can be used to show some desirable properties such as: (1) *convex hull property*: any point in the range $\mathbf{T}(t,s,r)$ is always inside the convex hull determined by \mathbf{b}_{ijk} ; (2) *affine invariance*: evaluating $\mathbf{T}'(t,s,r)$ from the affinely transformed points $\Phi(\mathbf{b}_{ijk})$ is the same as applying the affine transformation Φ to $\mathbf{T}(t,s,r)$ that was evaluated from the original points \mathbf{b}_{ijk} ; (3) *Invariance under affine parameter transformations*: we may affinely transform the cube in the domain (doing scaling or translation, for instance) and still obtain the same volume determined by the points in the range \mathbf{b}_{ijk} . Other interesting properties can be seen in Farin [9], where the trilinear transformation is given as a triple tensor product of Bézier curves of degree 1.

Trilinear transformations are also used in popular algorithms in C.G. such as Marching Cube (Foley et al. [10], Watt et al. [10]) and Levoy's color and opacity computation (Levoy [12]).

We are now entitled to deform tomographic models, which are represented as scalar fields $\mathbf{I}:\mathbf{R}^3 \rightarrow \mathbf{R}$ for the original model and $\mathbf{I}_I:\mathbf{R}^3 \rightarrow \mathbf{R}$ for the deformed one. These scalar fields assign a color intensity to a point in the volume, typically a value between 0 and 255 (1 byte). Since we are given 2D images of gray levels, we could assign to a point (t,s,r) in the volume the gray level associated with pixel (t,s) in the r -th image.

The deformation scheme assumes the existence of a cube large enough to contain the stack of images, represented by the scalar field \mathbf{I} . Thus, if we know the original model's scalar field we can obtain the deformed model's scalar field by doing $\mathbf{I}_I(\mathbf{T}(t,s,r)) = \mathbf{I}(t,s,r)$. To avoid sampling problems that can happen to highly deformed models we invert the scheme: we enclose the original model in the volume determined by the points \mathbf{b}_{ijk} (in the space regarded originally as the range). We find the deformed model by doing: $\mathbf{I}_I(t,s,r) = \mathbf{I}(\mathbf{T}(t,s,r))$. That means that the deformed model will appear inside the unit cube. This scheme is called a *re-sampling filter* (Wolberg [13]).

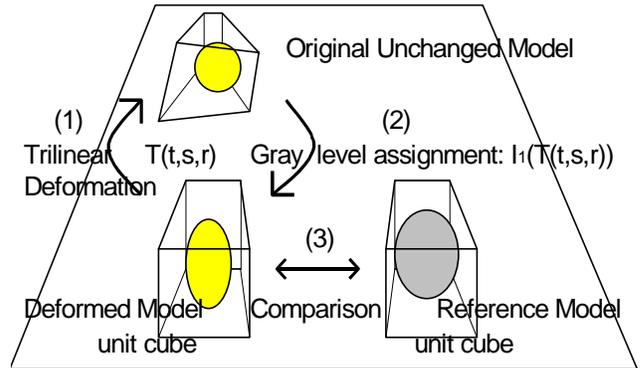


Figure 2 Iterative Deformation of a 3D model.

We illustrate our iterative deformation algorithm in Figure 2. We first enclose the 3D model in the volume determined by the points \mathbf{b}_{ijk} (step 1); we then scan the unit cube to visit each voxel (t,s,r) to trilinearly map it into the voxel $\mathbf{T}(t,s,r)$ in the volume determined by the points \mathbf{b}_{ijk} and then we obtain (t,s,r) 's gray level by doing $\mathbf{I}_I(t,s,r) = \mathbf{I}(\mathbf{T}(t,s,r))$ (step 2); finally we compare in a voxel by voxel basis the deformed model with the reference model (which is placed in another unit cube and has scalar field $\mathbf{I}_2:\mathbf{R}^3 \rightarrow \mathbf{R}$) (step 3). We will see how these comparisons are done in a moment. To address the

variability that exists in the brain’s local structures we need to apply subdivisions to the volumes. By finding the midpoint of each edge, the center of each face and the centroid of the volume, and then connecting them appropriately we obtain eight new volumes. These subdivisions are done to all three volumes in Figure 2. We may now apply a trilinear map to each sub-volume in the volume determined by \mathbf{b}_{ijk} to obtain as result the corresponding sub-volume in the unit cube. We then proceed with the comparison with the corresponding sub-volume in the reference model. The validity of these operations is guaranteed by the 3 properties discussed earlier. The volumes in Figure 2 can be seen as corresponding sub-volumes from the three initial volumes that contain the models.

The comparisons use a functional called *energy* which is based on a voxel by voxel statistical measure called *correlation* of gray levels, and it is given by:

$$e(\mathbf{I}_1, \mathbf{I}_2) = 1 - |\text{cov}(\mathbf{I}_1, \mathbf{I}_2) / \sigma_1 \cdot \sigma_2| \quad (\text{Eq. 1})$$

where *cov* is the covariance between the two intensity-based models, given by:

$$\text{cov}(\mathbf{I}_1, \mathbf{I}_2) = \mathbf{E}[(\mathbf{I}_1 - \mu_1)(\mathbf{I}_2 - \mu_2)]$$

with \mathbf{E} being the expected value, μ_1, μ_2 , the simple average of gray levels of each unit cube and σ_1, σ_2 , their standard deviation.

Roughly speaking, we can say that the larger the energy, the more distinct the two models are from each other. There are some technical limitations on this claim, that can be seen in Melo [14]. However, the experiments have shown that the use of this functional results in a superior quality of matching when compared with the use of a simple voxel by voxel absolute value of the differences of gray levels.

It becomes clear now that we need to generate a sequence of configurations for the vertices \mathbf{b}_{ijk} in such a way to minimize the energy between the two models. Due to the complexity of this functional and the variability present in different brain models, there is no recipe for reaching the absolute minimum. We use an optimization algorithm called *simulated annealing* (Metropolis et al. [15], Laarhoven et al. [16], Melo [14]), with an improvement that takes advantage of the energy’s smoothness: *gradient descent* technique (Wismer et al.

[17]). The algorithm starts off with an arbitrary configuration and computes the energy’s gradient at that configuration; the next configuration is computed as the current configuration added with the negative of the current gradient vector (multiplied by a scalar). When we reach a local minimum (not able to descend) we record this position and then “move away” from the local minimum by accepting a sequence of configurations with increasing energy, until we step on the top of a “hill” to explore other valleys. Laarhoven et al. [16] shows that, although there is no guarantee of reaching the absolute minimum, it actually does converge asymptotically to the optimal solution. To us that means the more the algorithm iterates the better the quality of the matching.

We may now announce the standard iterative algorithm which comprises two phases: the normalization of the entire volume and the normalization of sub-volumes. For the first phase we optimize the positioning of the eight vertices and, in order to compute the energy of a given configuration of vertices, we apply the 3 steps associated with Figure 2 for the deformation and use the resulting scalar field \mathbf{I}_1 together with the given scalar field \mathbf{I}_2 in (Eq. 1). Notice that a configuration here is an element of a 24-dimensional space (eight vertices with three coordinates: \mathbf{x}, \mathbf{y} and \mathbf{z}).

When we reach the best configuration in phase one, we subdivide the volume corresponding to this configuration and also the unit cubes for the deforming model and for the reference. In this phase we only optimize the centroid (affecting each sub-volume in just one vertex). In order to compute the energy of a given configuration we apply the 3 steps associated with Figure 2 for the deformation independently for each triple of sub-volumes: in the \mathbf{b}_{ijk} -volume, its corresponding one in the unit cube for deforming model and in the reference unit cube. The energy associated with a configuration is the summation of the energy of the eight sub-volumes. When we reach a satisfactory energy level we subdivide each sub-volume and reapply the algorithm to each of them. We recursively repeat this procedure until no more subdivisions can be done. We should point out that no subdivision is required to a sub-volume when its energy reaches a satisfactory value before the same happens to this sub-volume’s siblings.

4 Improvements in Speed and Quality

The more experienced reader may have noticed the problem that appears due to the lack of smoothness presented by the scalar field of the deformed model on the borders between sub-volumes. The visual results are the presence of objectionable kinks and artifacts on otherwise

smooth areas such as the surface of the cortex or cerebellum (for more details: Melo [14]). We can satisfactorily minimize these artifacts by reapplying the algorithm to some suitably composed volumes. We can do that since at the end of our algorithm we are left with the set of cells at the highest subdivision level. What we need to do is to compose a sub-volume from eight contiguous cells in such a way that its centroid lies on a surface that used to be a border between two old subdivided volumes, optimized in the second phase of the algorithm. We then apply the algorithm normally to this composed volume. This artifact correction can be done before each new subdivision in phase 2 takes place.

In view of the computational effort necessary for the optimization we decided to speed up the algorithm by representing the models in a multi-resolution fashion. We adopted a tensor product of three wavelets spaces using Haar basis to represent the scalar field of each model (Muraki [18]). A pyramid of 3-dimensional models in different resolutions is built at the beginning of our algorithm. Each time we have a volume or a subdivided volume to optimize, we take its representation in the lowest resolution and optimize it first. When we reach the best possible energy value, we transform the last configuration to the next higher resolution and then optimize it further in this resolution. When we finish in this resolution, we repeat this procedure until we reach the highest resolution. Only then we proceed to the next subdivided volume.

The idea of using a multi-resolution representation is to achieve a more controllable gradual convergence. In the lowest resolution we may work with a large space of configurations; with that we may obtain a rough convergence. As we move to a higher resolution we may gradually restrict the space of configurations, perhaps centered in the configurations reached in previous resolutions. We usually save time by doing this, but we should not expect a better quality in the matching, on the contrary, normally the result is worst than if we had used that initial space of configurations in the highest resolution in the first place (although much faster).

Another improvement is to take advantage of this algorithm's intrinsic parallel structure. In our implementation we use PVM (Parallel Virtual Machine) to distribute the computation among eight processors (Al Geist et al. [19]). The distribution obeys a master-slave structure. We leave the first phase and the optimization of the centroid in the subdivided volume to the master processor and after that we assign a sub-volume to each of the slaves. They take over from then on. As for the inter-slave communication, it is necessary only because of the

artifact correction that requires neighboring slaves to exchange information concerned the borders (Melo [14]). The improvement achieved with this distribution is remarkable. Figure 3 presents a flowchart of a slave task. By current position we mean the geometric location in the space of the energy graph (or simply the current configuration). Notice that a local minimum is reached when the gradient vector has zero norm. A satisfactory value for the energy is determined by the user's arbitration for a set of control parameters; the current local minimum is only recorded if it is lower than the best local minimum so far.

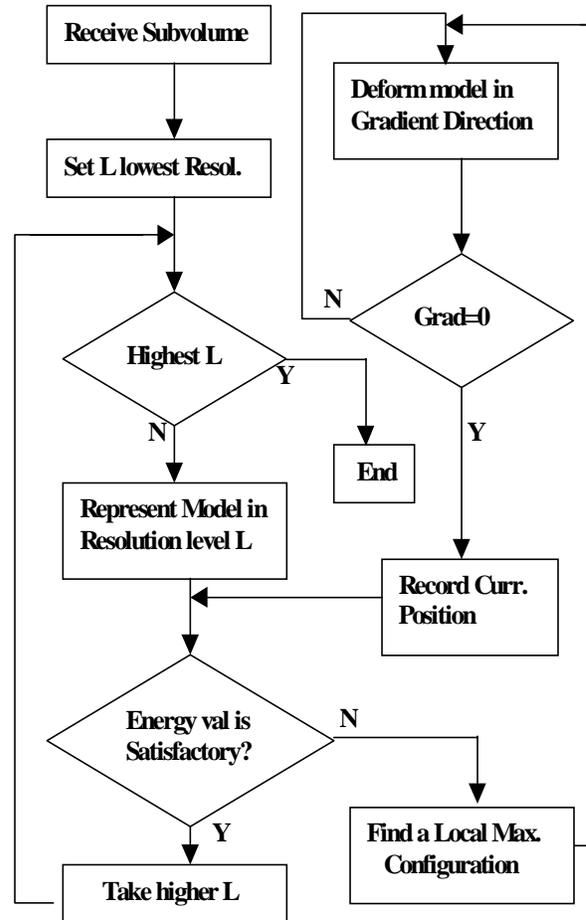


Figure 3 Flowchart of a slave task.

5 Results

Serial and parallel versions of an iterative piecewise trilinear deformation based on the aforementioned algorithm were applied to two data sets with many parameter settings. All data sets were provided by the PET

center in the Phoenix Good Samaritan Hospital. We should point out that the only real 3D deformation approaches for normalization of human brain models available are simple affine deformations or piecewise affine deformations; the best nonlinear methods of today still work in 2D.

The brain models were semi-automatically segmented from the scalp and background; they are: two sets of 86 images of size 256x256 (source and target); and two sets of 100 images of size 256x256.

We used versions of the algorithm with the energy function e as well as the absolute differences in gray level. The results with the latter function were obtained faster by orders of magnitude, but with a severely inferior quality. A cost function that accounted for matching highly contrasting areas was added to that function, but it only improved the contour matching. The internal structures were normally damaged, and the results were discarded. Figures 4 and 5 concern data sets from two different subjects. For both we used 4 levels of subdivision and 3 levels of resolution. Higher numbers of resolution levels were attempted, with better performance but inferior results. In Figure 4 we have 2 columns, each with a set of triple images: a slice from the source (individual brain), the corresponding slice from the reference followed by the corresponding slice from the deformed brain model. We start in the left column, from the upper image to lower ones, and then we move on to the right column. This procedure corresponds to moving away from the base of the cerebellum up into the top of the brain.

In these figures, only a few central slices out of 86 are shown. The serial version spent 40h27min in a SGI O2 machine. The parallel version took about 10h in the same machine. The initial energy was 0.1555. An energy value of 0.0 means that the models are identical (up to global gray displacements). A value of 1.0 means that the models possess no linear correlation between themselves, with respect to any of their sub-regions. Since the energy function is based on the linear correlation coefficient, which is standard in many fields, these numbers can be directly tested and used in comparisons with other methods. With an initial bounding box identification, our energy value went down to 0.1235. After the optimization of the eight initial vertices, the value became 0.0942, and the final value was 0.0565. Needless to say, the quality of the deformation visually improved during these steps, but we only show the last.

The reader should notice the higher similarity of internal structures between the reference and the deformed models as opposed to that between the source and the

reference ones. It is more visible around the inter-hemispherical plane and close to the contours.

In Figure 5 we have two views of the contour surfaces for each model in the same order: source, reference and result. The first two images were done by projecting the models in a standard plane and illuminating them. The normals were computed by using the neighboring voxels and their elevation in relation to the plane. The light source was placed in the line of sight. The other two images are two planar cuts of each model (sagittal and coronal planes) in the same order: source, reference and result.

Figure 6 brings data sets from the same subject. The original model was initially deformed by using the popular package *SPM96* (Statistical Parametric Mapping) (Frackowiak et al [20]). The deformation comprises of a general affine transformation (automatic) followed by a discrete cosine transform, in order to express the intensity maps in terms of a smooth basis. Iterative approximations to a deformation field, which is derived from the models' differences, are conducted to obtain the final co-registered model. Each slice from the resulting deformation is shown as the first part in a triple image in each column. The second part is a slice of the original model, which is our reference model. Our aim is to 'undeform' the model deformed by *SPM96*. Our results are shown as the third part in the triple images. Before applying the algorithm, we rotated the model by 14 degrees around the X-axis (adding some artifacts). We used 4 levels of subdivision and 3 levels of resolution. Higher numbers of resolution levels were also attempted, with better performance but inferior results.

In Figure 6 only a few central slices out of 100 are shown. The serial version spent 70h30min in a busy SGI onyx machine (low priority-250). The parallel version in 8 machines takes approximately 1/8 of the serial version's time + few hours. The initial energy was 0.2365. With the bounding box identification, it went down to 0.1305. After the optimization of the eight initial vertices, the value became 0.1214, and the final value was 0.0566.

6 Conclusion

We have presented a nonlinear approach for the 3D normalization of human brain models. In view of the current approaches, which face computational problems, we have decided to introduce a method intended to be computationally simple and yet able to perform a 'better' 3D normalization than a simple linear method, which is nowadays the only method accepted to validate 3D normalization. Although a piecewise trilinear transformation obviously suffers from smoothness

problems, it still needed to be formulated, tested and validated before more sophisticated methods are attempted, for its simplicity and its relatively reasonable computational time.

Several aspects of this method may benefit from further investigation. For instance, smoother wavelets basis may decrease some artifacts. It also may induce a representation for a database of brain models, deformed or not, in the future. A study on how the energy function and the optimizer behave as we change resolution levels may be important. We need to know with certainty whether a convergence in lower level really provides a convenient starting point for a higher one. Our results suggest that in most cases, if we want to improve the quality, we need to use only higher resolution levels.

It might be necessary to incorporate more precision in the concept of quality of a matching. In some cases it might be convenient to have a good overall statistical matching rather than a good visual matching.

Finally, the method may need a strong statistical validation to become popular. It has been suggested that the method should be validated by deforming MRI models from a group of patients into a standard shape. The deformed models would be co-registered to PET images from the same patients in two situations: resting and performing an elementary task. Some statistical measures would indicate if the deformation improved the precision of how the regions corresponding to the elementary task are detected.

Acknowledgment

We would like to thank the contributions from Dr. A Rockwood, E. Reiman, Phoenix Good Samaritan Hospital and Arizona State University, and the financial support from the CNPq and Robert S. Flinn Foundation of Arizona.

References

[1] J. Tairach and P.Tournoux. *Coplanar stereotaxic atlas of the human brain*. T. Verlag, Stuttgart-NY88

[2] Collins, D. L., Neelin, P., Peters, T. M. and Evans, E. C. *Autom. 3D Intersubject Reg. Of MR Vol. Data in Std Tail. Space*. J. of C. Assisted Tomography-18,- '94.

[3] Sederberg, T.W., and Parry, S. R. *Free Form Deform. of S. Geom. Models SIGGRAPH'86*. 20(4)- '86

[4] Minoshima, S., Koeppe, K., Frey, A. and Kuhl, D. E. *Anatomic Std: Linear Scaling and Nonlinear Warping of Functional Brain Images*. J. of Nuclear Medicine 9- '94

[5] Ping, N. *Deformation Using Scattered Landmark Points*. Master's Thesis, Arizona State University, '95.

[6] Yun, L. S. *Scattered Data Mod. and Deform. with Transfinite Constr*. Ph.D. Thesis, Arizona St. Univ. '96

[7] Miller, M., Christensen, G., Amit, Y. and Grenander, U. *Math. Textbook of Deform. Neuroan.* Proceedings of the N. A. S., 90(24)- '93.

[8] Terzopoulos, D., Witkin, A. and Fleischer, K. *Elastically Deform. Models. SIGGRAPH'87* 20(4) '97

[9] Farin, G. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*. Academic Press, New York, 3rd Edition, '90.

[10] Foley, J. D., van Dam, A. Feiner, S. K. and Hughes, J. F. *Computer Graphics: Principle and Practice*. 2nd Ed. Addison Wesley - '91

[11] Watt, A. and Watt, M. *Advanced Animation and Rendering Techniques, Theory and Practice*. Addison Wesley - '92.

[12] Levoy, M. *Volume Rendering: Display of Surfaces from Volume Data*. IEEE CG & A- '88

[13] Wolberg, G. *Digital Image Warping*-IEEE Computer Society Monograph-'92

[14] Melo, S. de B., *Nonlinear Normalization of Brain Models Using Piecewise Trilinear Deformation*-Ph.D. thesis-Arizona State University-'98

[15] Metropolis, N., Rosebluth, A., Rosebluth, M., Teller, A. and Teller, E. *Equation of State Calculations by fast Computing Machines*. Journal of Chemical Physics, 21-'53.

[16] Laarhoven, P. J. M. and Aarts, E. H. L. *Simulated Annealing: Theory and Appl.s* D. Reidel Publ. Co., '87.

[17] Wismer, D. A., and Chattergy, R. *Introduction to Nonlinear Optimization: A Problem Solving Approach*. North Holland-New York, '78.

[18] Muraki, S. *Volume Data and Wavelets Transforms*. IEEE Computer Graphics & Appl.- '93.

[19] Geist, A.L., Beguelin, A., Dongarra, J., Jiang, W., Manchek, R. & Sunderam, V. *PVM: Parallel Virt. Mach.: A user's guide & Tutorial for Networked Parallel Comp.* MIT Press-'94.

[20] Frackowiak, R.S.J., Friston, K.J., Frith, C.D., Dolan, R. J. and Mazziotta, J.C. *Human Brain Function*. Academic Press-'97.

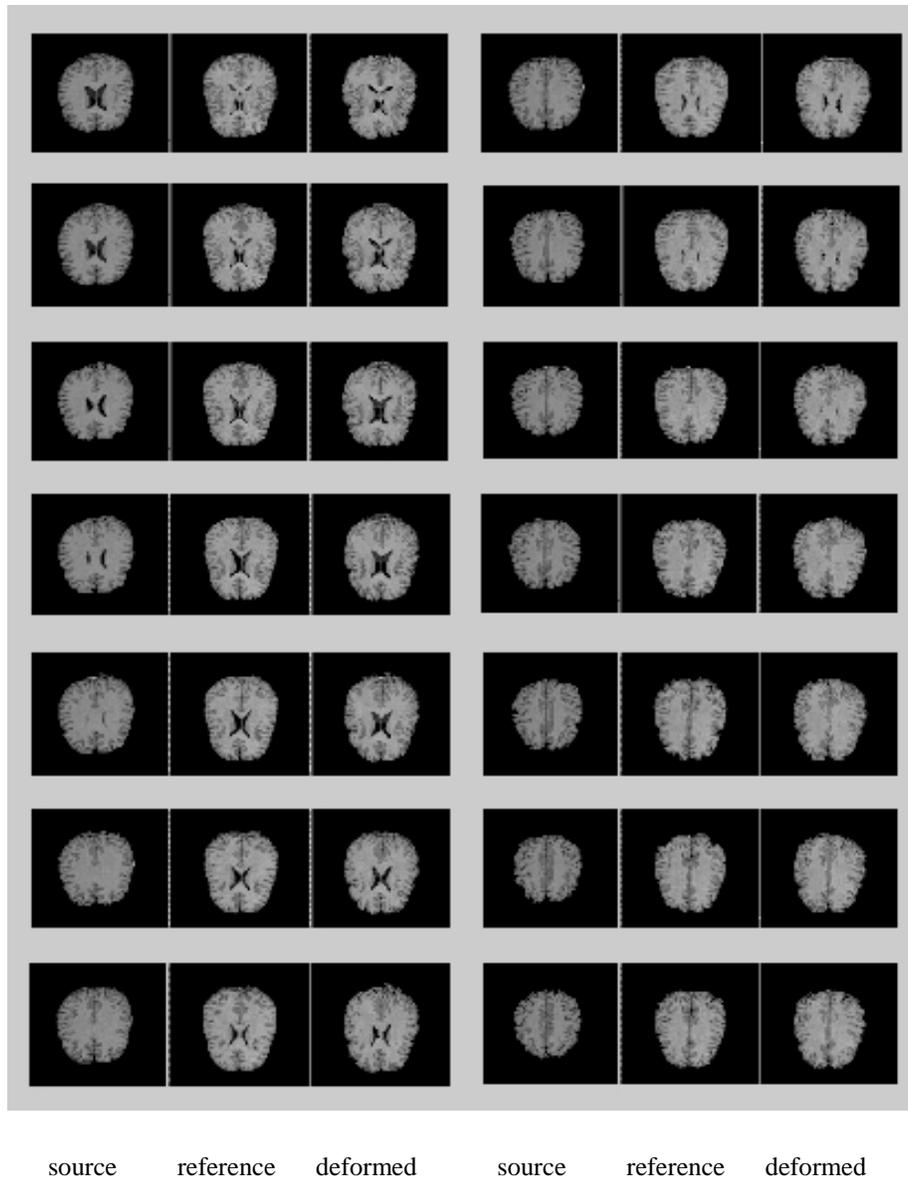


Figure 4 Results of a Piecewise Trilinear Deformation (14 central slices)

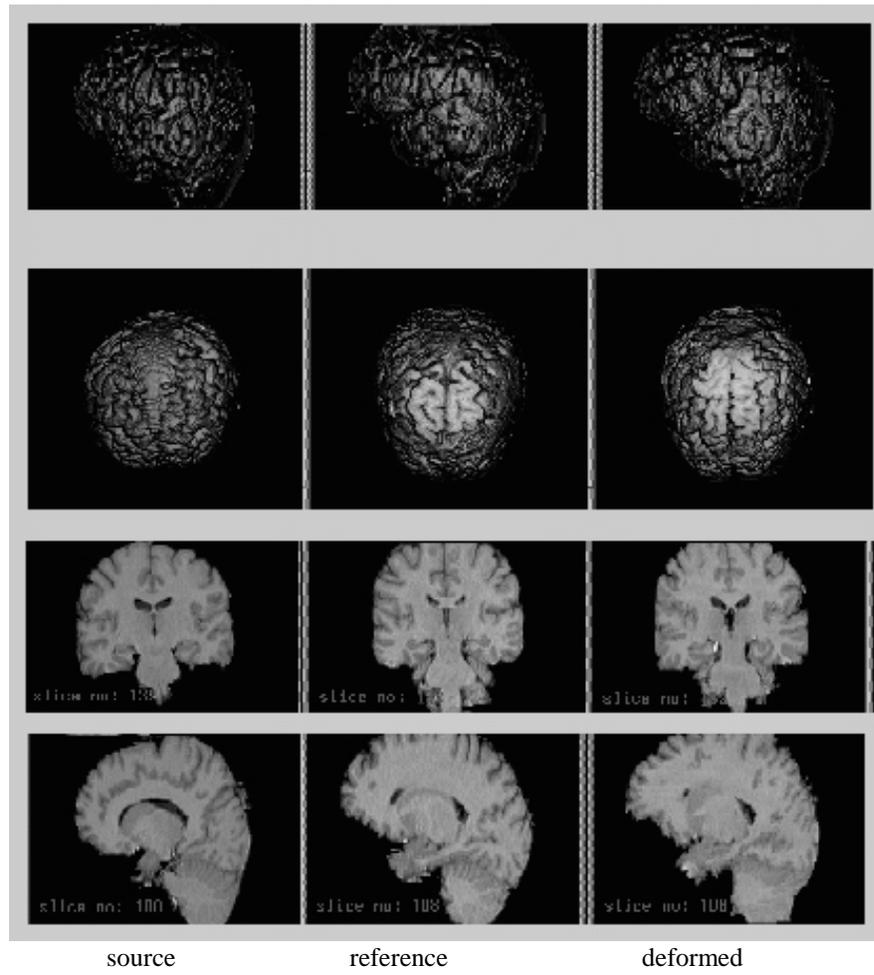


Figure 5 Other Views of the Deformation in Figure 4

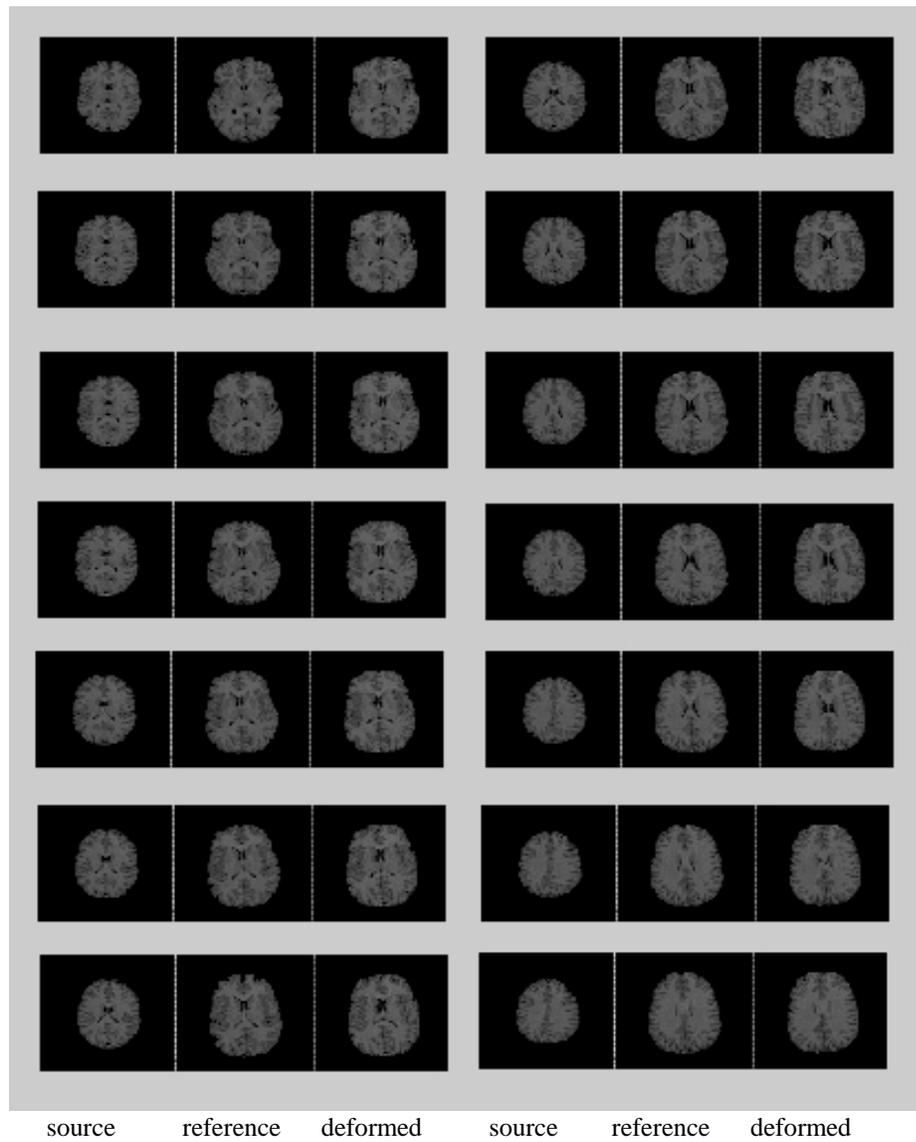


Figure 6 Other Results of a Piecewise Trilinear Deformation (14 central slices)