Atlases

Automatic Retrieval of Anatomical Structures in 3D Medical Images

Jérôme Declerck, Gérard Subsol
Jean-Philippe Thirion and Nicholas Ayache

INRIA, B.P. 93, 06 902 Sophia Antipolis Cedex, France
E-mail: Jerome.Declerck@sophia.inria.fr
http://zenon.inria.fr:8003/epidaure/Epidaure-eng.html

Abstract. This paper describes a method to automatically generate the mapping between a completely labeled reference image and 3D medical images of patients. To achieve this, we combined three techniques: the extraction of 3D feature lines, their non-rigid registration and the extension of the deformation to the whole image space using warping techniques. As experimental results, we present the retrieval of the cortical and ventricles structures in MRI images of the brain.

1 Introduction

It becomes needless to emphasize the advantages of electronic atlases versus conventional paper atlases. However, even if such atlases are available [9] and even if Computer Graphic techniques are sufficiently developed to manipulate and display those atlases in real time, there remains a crucial need for *automatic* tools to analyze the variability of features between patients [11], [14]. For that purpose, we have to find correspondences between the image of any patient and the atlases.

This paper presents one possible approach to achieve this goal, usually referred to as a segmentation problem, by using a strong a priori knowledge of the human anatomy.

There are usually two complementary ways to explore, which are the region based technique using the voxel values inside the regions [3], and the feature based one only using the boundaries of those regions [5], such as the interfaces between organs, or specific lines or points of those surfaces [17], [16]

In the present paper, we concentrate on a feature line based technique to segment fully automatically the same organ in the images of different patients. We give first a global description of the method, which is then detailed into feature lines extraction using differential geometry, registration of lines using deformable models, and at last 3D space deformation using warping techniques. Finally, we present a practical example, which is the automatic extraction of the cortical and ventricle surfaces from the 3D MRI images of a patient.

2 A global description of the method

Let us assume that we have a 3D reference image I_r and an associated fully labeled image M_r, called map (figure 1) in which each voxel value specifies the

type of a corresponding structure in I_r. We call *structure* a set of connected voxels of M_r having the same label. I_p is the image of a new patient to process.

We suppose also that images I_r and I_p have been acquired with the same modality and parameter settings: their intensities are very similar.

To find the correspondence between I_p and the reference map M_r, we propose to follow this scheme (figure 1):

- automatically find and label some features in I_r corresponding to a labeled structure S_r of M_r,
- automatically find the equivalent features in I_p,
- find the correspondence $C_{p,r}$ between those features,
- either deform individual structures or the global map M_r into new structures or into a new map M_p, which is exactly superimposable to I_p. This step is achieved by finding a space warping function $D_{p,r}$ and applying it to M_r. We retrieve then the structure S_p in I_p corresponding to S_r in I_r.

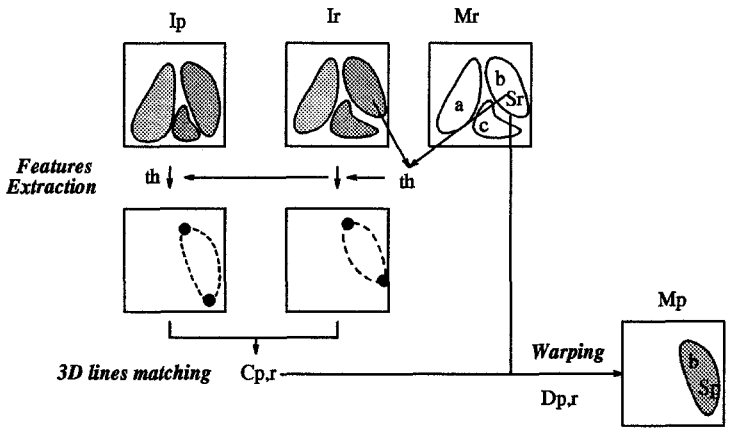

Fig. 1. *Features of the labeled structure S_r are extracted from the reference image I_r. Equivalent features are extracted from I_p. By finding the registration $C_{p,r}$ between these features, a warping function $D_{p,r}$ can be computed to obtain the map M_p superimposable on I_p and so, the structure S_p.*

3 Feature based non-rigid registration

3.1 The feature type: the crest lines

Raw medical images are stored in a discrete 3D matrix $I = f(x, y, z)$. By thresholding $I = I_0$, isosurfaces of organs are computed (for instance the brain with MRI images). To get a sparser but relevant representation of the isosurface, we propose to use the "crest lines" introduced in [10].

They are defined as the successive loci of a surface for which the largest principal curvature is locally maximal in the principal direction (see figure 2,

left). Let k_1 be the principal curvature with maximal curvature in absolute value and $\overrightarrow{t_1}$ the associated principal direction, each point of a crest line verifies: $\overrightarrow{\nabla} k_1 . \overrightarrow{t_1} = 0$. These lines are automatically extracted by the "marching lines" algorithm [17].

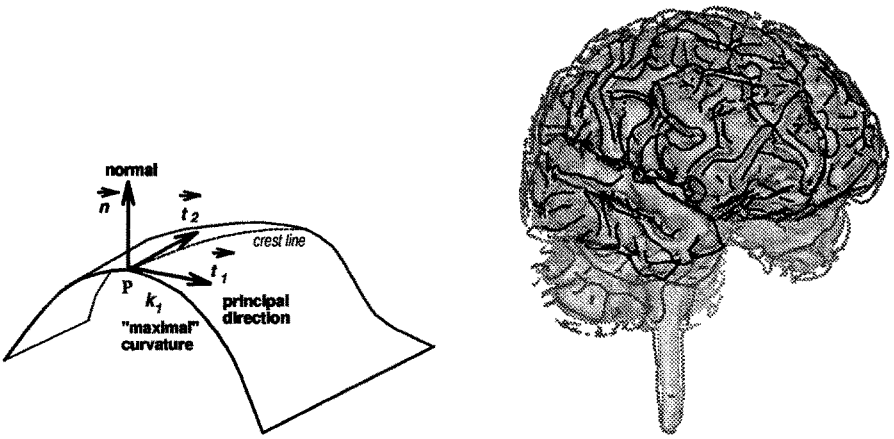

Fig. 2. *Left: mathematical crest line definition. Right: crest lines on a brain.*

In fact, crest lines turned out to be anatomically meaningful. Thus, in figure 2, right, crest lines follow the cortical convolutions, emphasizing the sulci and gyri patterns described by Ono et al. [12].

3.2 The crest lines extraction

We want to extract the crest lines of the structures S_r in I_r and their equivalent in I_p. The problem is then to find the threshold I_0. I_0 is representative of the interface between S_r and the other structures and can be computed as the mean value of the voxels of I_r labelled as S_r in M_r. As the images have the same dynamic, we can use the same threshold in I_p.

3.3 The 3D non-rigid registration algorithm

The 3D lines registration algorithm is a key point in our scheme: given two sets A and A' composed of the crest lines L_i and L'_j extracted from two different images, we want to find which lines L_i of A correspond to which lines L'_j of A'. Two difficulties arise: the number of lines of each set is quite large (several hundreds to more than a thousand) and the registration between A and A' is not rigid, preventing from using Euclidean invariant based methods as in [8].

Zhang [18] and independently Besl [1] introduced an "iterative closest points" matching method. Both authors use this algorithm to register free-form curves but only for the rigid case. Nevertheless, we can improve and generalize this

method to our problem and our algorithm follows the steps of the "iterative closest point" method.

Points matching Each point of A is linked with its closest neighbour in A' according to the Euclidean distance. We plan also to use in the distance computation the differential curve parameters as the tangent, normal, curvature and torsion [8] or surface parameters as the normal, the principal directions and principal curvatures as described in [6]. This gives a lists of registered points, $\mathcal{C}_1$.

But as we have curves, i.e. an *ordered* list of points, we can apply some topological constraints in order to remove non-consistent couples of registered points and then to avoid irrelevant configurations. We obtain an other list $\mathcal{C}_2$.

From $\mathcal{C}_2$, two coefficients can be computed: p_i^j and $p_j'^i$ which are the proportion of the curve i of A matched with the curve j of A' and vice versa. By thresholding, $p_i^j \geq thr$ and $p_j'^i \geq thr$, we can determine the curves which are "registered" at thr percent. For instance, curves can be considered completely registered when $p_i^j \geq 0.5$ and $p_j'^i \geq 0.5$.

Least-squares transformation We want to register A and A' with polynomial transformations. The 0^{th}-order is a rigid transformation and 1^{st}-order an affine one but they are not sufficient for an accurate non-rigid registration. So, we use 2^{nd}-order polynomial transformations defined by (for the x coordinate):

$$x' = a_1x^2 + a_2y^2 + a_3z^2 + a_4xy + a_5yz + a_6xz + a_7x + a_8y + a_9z + a_{10}$$

As these polynomials are linear in their coefficients, we can use the classical least-squares method to compute the a_i from $\mathcal{C}_2$.

2^{nd}-order polynomial transformations give accurate registration but we are not able to decompose them into intuitive physical meaning transformations such as rotation, translation or scaling. Notice that, at each iteration, we compose the transformation with a 2^{nd}-order polynomial and so, we obtain after n iterations, a potential 2^n-order polynomial transformation.

2^{nd}-order transformations are also used by Greitz et al. [7] to model natural deformations as brain bending.

Updating The transformation is then applied and the algorithm iterates again or stops according to some criteria (mean value of the distance distribution between matched points, stability of the registration coefficients p_i^j and $p_j'^i$, threshold on the matrix norm $||T - I_d||$ where T is the transformation and I_d the identity matrix).

Parameters adaptation By incrementing the threshold value thr at each iteration and by taking only into account the matched point couples belonging to "registered" curves at thr percent, the algorithm tends to improve the registration of already matched curves and to discard isolated ones. Moreover, we can

begin to apply rigid transformations to align the two sets of lines, then affine transformations to scale them and, at last, quadratic transformations to refine the registration.

At the end, we obtain a good registration between the two sets of lines and so, a point to point correspondence between lines. These correspondences give significant landmarks to define a B-spline based warping on the whole space.

4 Space warping

4.1 The problem

Let us call F, the exact matching function, i.e. the geometric transformation that takes a point P_r in I_r and gives its anatomically equivalent P_p in I_p:

$$\begin{aligned} F \ : \quad & I_r \longrightarrow I_p \\ & P_r(x,y,z) \longmapsto P_p(u,v,w) \end{aligned}$$

The set of matched points $\mathcal{M}$ obtained with the feature based registration is an estimation of some pairs $(P_r, F(P_r))$. We have then a sparse estimation of F which we can extend to the whole space by a warping function $\varphi_{\mathcal{M}}$.

4.2 Calculation of the warping function

Bookstein and Green [2] define $\varphi_{\mathcal{M}}$ as a thin-plate spline interpolating function but only in 2D. Two problems raise if we apply this definition to our approach. First, it appears difficult to generalize in 3D. Second, interpolation is relevant when the matched points of $\mathcal{M}$ are totally reliable and regularly distributed (for example, a few points manually located). In our case, these points are not totally reliable due to possible mismatches of the registration algorithm and are sparsed in a few compact areas as they belong to lines.

So for $\varphi_{\mathcal{M}}$, we prefer to use an approximation function which is regular enough to minimize the influence of erroneous matched points. We choose a B-spline tensor product which is easy to define and control in 3D.

The B-spline approximation We define the coordinate functions of $\varphi_{\mathcal{M}}$, (u,v,w) as a three-dimensional tensor product of B-spline basis functions. For instance, for u:

$$u(x,y,z) = \sum_{i=0}^{n_x-1} \sum_{j=0}^{n_y-1} \sum_{k=0}^{n_z-1} \alpha_{ijk}\, B^x_{i,K}(x)\, B^y_{j,K}(y)\, B^z_{k,K}(z)$$

with the following notations:

- n_x : the number of control points in the x direction. n_x sets the accuracy of the approximation.
- α : the 3D matrix of the control points abscissae.

- $B^x_{i,K}$: the i^{th} B-spline basis function. Its order is K. u is then a piecewise K^{th} degree polynomial in each variable x, y and z, easy to evaluate with the de Casteljau algorithm [13]. We choose cubic B-splines in our examples ($K = 3$), for their regularity properties. For the B-spline knots, we take the classic regular mesh.

For a given number of control points and a set of B-spline basis functions, u is completely defined by the α_{ijk}. They are calculated by minimizing a criterion computed with the set $\mathcal{M}$ of matched points.

The criterion We define three criteria J^x, J^y and J^z (one for each coordinate) to determine the best $\varphi_{\mathcal{M}}$, with respect to our data. For instance, for u, J^x splits in two parts:

- position term. For each data point P_r, $u(P_r)$ must be as close as possible to the abscissa of P_p. We choose a least-squares criterion:

$$J^x_{position}(u) = \sum_{l=1}^{N} \left(u(x^l_1, y^l_1, z^l_1) - x^l_2\right)^2$$

- smoothing term. B-splines have intrinsic rigidity properties, but it is sometimes not enough. We choose a second order Tikhonov stabilizer: it measures how far from an affine transformation the deformation is.

$$J^x_{smooth}(u) = \rho_s \int_{\mathbb{R}^3} \left[\frac{\partial^2 u}{\partial x^2}^2 + \frac{\partial^2 u}{\partial y^2}^2 + \frac{\partial^2 u}{\partial z^2}^2 + \frac{\partial^2 u}{\partial x \partial y}^2 + \frac{\partial^2 u}{\partial x \partial z}^2 + \frac{\partial^2 u}{\partial y \partial z}^2 \right]$$

where ρ_s is a weight coefficient. It is currently manually defined and some solutions to choose it automatically are under study.

The criterion to minimize is then: $J^x(u) = J^x_{position}(u) + J^x_{smooth}(u)$

The linear systems J^x is a positive quadratic function of the α_{ijk} variables. To find the coefficients that minimize J^x, we derive its expression with respect to all the α_{ijk}: it gives $n_x \times n_y \times n_z$ linear equations. Assembling those equations, we get a sparse, symmetric and positive linear system. We solve the 3 systems (one for each coordinate) to completely calculate $\phi_{\mathcal{M}}$.

5 Results and discussion

We apply the described process to retrieve the cortical and ventricles structures in MRI images of the brain of a patient from a reference segmented image (see figures 3 to 7).

The structures have been correctly recovered in spite of the sparse representation of the data. It shows that crest lines turn out to be significant features with few ambiguities and to be anatomically consistent. Moreover, a data point

has a *local* influence: to evaluate a transformed point, we need only $(K+1)^3$ control points (to be compared with the $n_x \times n_y \times n_z$ that have been calculated). Hence, the influence of outliers is very local. At last, the regularity of the B-splines defined $\varphi_{\mathcal{M}}$ allows to get consistent correspondences everywhere in the image. In particular, we can retrieve inner structures of the brain, with a warping only based on its external surface features.

6 Conclusion and perspectives

The proposed method allows us to build fully automatically the maps associated to the 3D images of new patients, from manually designed maps of reference patients. It can be used to efficiently initialize 3D surface snakes if a more precise final segmentation of the organs is needed [4], [15].

We especially thank Dr Ron Kikinis from the Brigham and Woman's hospital, Harvard Medical School, Boston, for having provided the segmented image of the brain, and the MR images to analyse. We also thank Digital Equipment Corporation (External Research Contract) and the BRA VIVA european project who partially supported this research.

References

1. Paul J. Besl and Neil D. McKay. A Method for Registration of 3-D Shapes. *IEEE PAMI*, 14(2):239–255, February 1992.
2. F.L. Bookstein and W.D.K. Green. Edge Information at landmarks in medical images. *SPIE Vol.1808*, 1992.
3. Gary E. Christensen, Michael I. Miller, and Michael Vannier. A 3D Deformable Magnetic Resonance Textbook Based on Elasticity. In *AAAI symposium: Application of Computer Vision in Medical Image Processing*, pages 153–156, Stanford, March 1994.
4. Isaac Cohen, Laurent Cohen, and Nicholas Ayache. Using deformable surfaces to segment 3D images and infer differential structures. In *CVGIP : Image understanding '92*, September 1992.
5. Chris Davatzikos and Jerry L. Prince. Brain Image Registration Based on Curve Mapping. In *IEEE Workshop on Biomedical Image Analysis*, pages 245–254, Seattle, June 1994.
6. Jacques Feldmar and Nicholas Ayache. Rigid and Affine Registration of Smooth Surfaces using Differential Properties. In *ECCV*, Stockholm (Sweden), May 1994. ECCV.
7. Torgny Greitz, Christian Bohm, Sven Holte, and Lars Eriksson. A Computerized Brain Atlas: Construction, Anatomical Content and Some Applications. *Journal of Computer Assisted Tomography*, 15(1):26–38, 1991.
8. A. Guéziec and N. Ayache. Smoothing and Matching of 3-D Space Curves. In *Visualization in Biomedical Computing*, pages 259–273, Chapel Hill, North Carolina (USA), October 1992. SPIE.

9. K. Höhne, A. Pommert, M Riemer, T. Schiemann, R. Schubert, and U. Tiede. Framework for the generation of 3D anatomical atlases. In R. Robb, editor, *Visualization in Biomedical Computing*, volume 1808, pages 510–520. SPIE, 1992. Chapell Hill.
10. Olivier Monga, Serge Benayoun, and Olivier D. Faugeras. Using Partial Derivatives of 3D Images to Extract Typical Surface Features. In *CVPR*, 1992.
11. Chahab Nastar. Vibration Modes for Nonrigid Motion Analysis in 3D Images. In *Proceedings of the Third European Conference on Computer Vision (ECCV '94)*, Stockholm, May 1994.
12. Michio Ono, Stefan Kubik, and Chad D. Abernathey. *Atlas of the Cerebral Sulci.* Georg Thieme Verlag, 1990.
13. J.-J. Risler. *Méthodes Mathématiques Pour la CAO.* Masson, 1991.
14. Gérard Subsol, Jean-Philippe Thirion, and Nicholas Ayache. Steps Towards Automatic Building of Anatomical Atlases. In *Visualization in Biomedical Computing '94*, October 1994.
15. D. Terzopoulos, A. Witkin, and M. Kaas. Constraints on deformable models : recovering 3D shape and non rigid motion. In *AI J.*, pages 91–123, 1988.
16. J-P Thirion. Extremal Points : definition and application to 3D image regist ration. In *IEEE conf. on Computer Vision and Pattern Recognition*, Seattle, June 1994.
17. J.P. Thirion and A. Gourdon. The Marching Lines Algorithm : new results and proofs. Technical Report 1881, INRIA, March 1993. to be published in CVGIP.
18. Zhengyou Zhang. On Local Matching of Free-Form Curves. In David Hogg and Roger Boyle, editors, *British Machine Vision Conference*, pages 347–356, Leeds (United Kingdom), September 1992. British Machine Vision Association, Springer-Verlag.

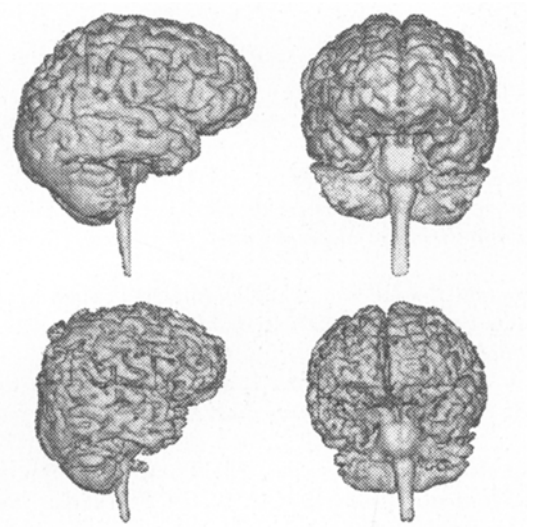

Fig. 3. *The data : top, the reference brain, bottom, the patient brain. Notice the differences in shapes and orientations.*

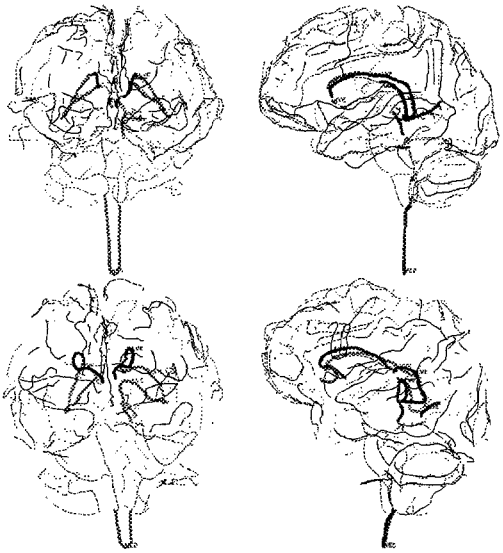

Fig. 4. *The crest lines have been automatically extracted from the reference brain (top) and from the patient brain (bottom). Some of the crest lines have been labeled (ventricles and medulla) and thickened for a better visualization.*

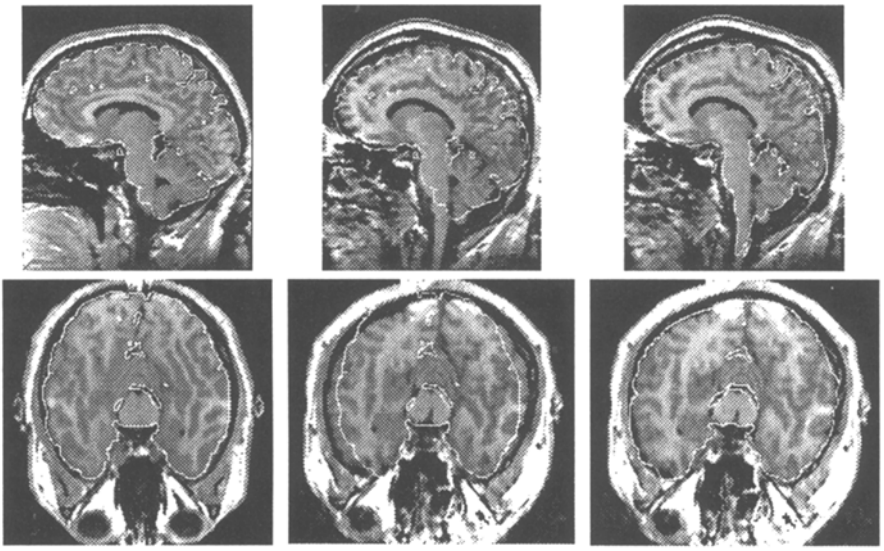

Fig. 5. *Left, the reference image I_r with the cortical surface S_r. Middle, the patient image I_p with S_r before deformation. Right, I_p with the result S_p of the found deformation applied on S_r.*

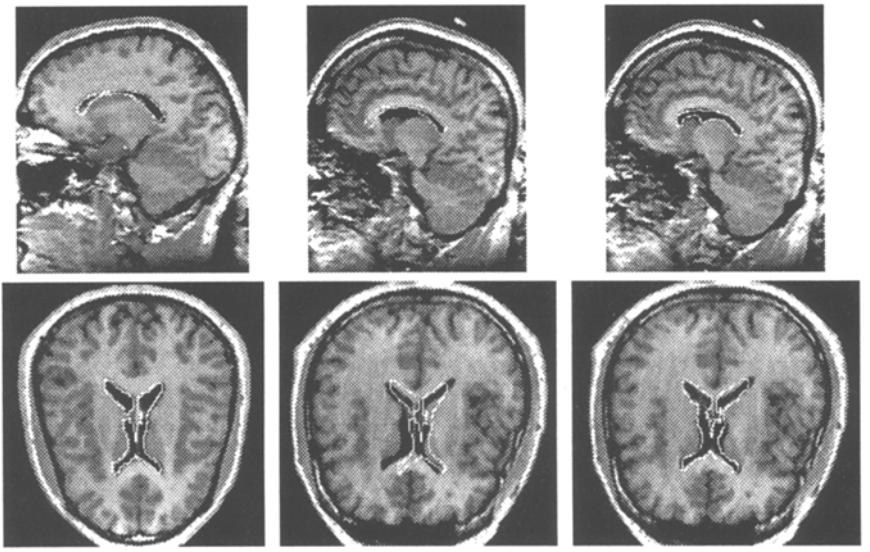

Fig. 6. *The same as figure 5, with the ventricles surface.*

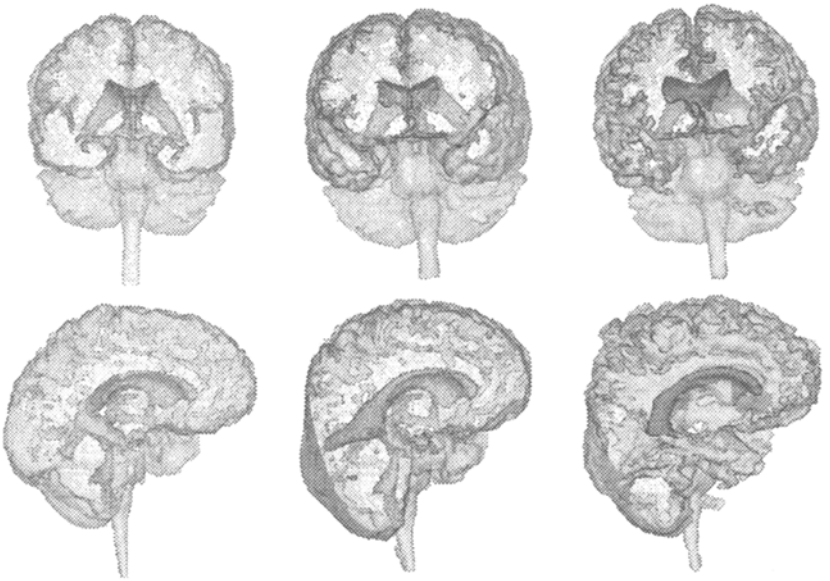

Fig. 7. *Different 3D views of the brains with their ventricle: left, the reference brain, middle, the same after warping and right, the patient brain.*