Area Preserving Cortex Unfolding

Jean-Philippe Pons[1], Renaud Keriven[1], and Olivier Faugeras[2]

[1] CERTIS, ENPC,
Marne-La-Vallée, France
Jean-Philippe.Pons@sophia.inria.fr
[2] Odyssée Lab, INRIA,
Sophia-Antipolis, France

Abstract. We propose a new method to generate unfolded area preserving representations of the cerebral cortex. The cortical surface is evolved with an application-specific normal motion, and an adequate tangential motion is constructed in order to ensure an exact area preservation throughout the evolution. We describe the continuous formulation of our method as well as its numerical implementation with triangulated surfaces and level sets. We show its applicability by computing inflated representations of the cortex from real brain data.

1 Introduction

Building unfolded representations of the cerebral cortex has become an important area of research in medical imaging. On a simplified geometry, it becomes easier to visualize and analyze functional or structural properties of the cortex, particularly in previously hidden sulcal regions. Moreover, if some metric properties of the cortical surface can be preserved while eliminating folds, it makes sense to map the data from different subjects in a canonical space for building brain atlases.

Three types of unfolded reprentations have been proposed in the litterature: inflated, that is to say a smoothed version of the cortex retaining its overall shape, spherical and flattened (see [1] and references therein). Three metric properties have been considered: geodesic distances, angles and areas. Unfortunately, preserving distances exactly is impossible because the cortical surface and its simplified version will have different Gaussian curvature [2]. Moreover, it is not possible to preserve angles and areas simultaneously.

As a consequence, early methods for cortex unfolding have settled for variational approaches, leading to a variety of local forces encouraging an approximate preservation of area and angle while smoothing the surface [3,4]. In [1], the authors point out that the result of such methods are not optimal with respect to any metric property, and work out a method which focuses on distances only and minimizes their distortion.

A lot of work has focused on building conformal representations of the cortical surface, i.e. one-to-one, onto, and angle preserving mappings between the cortex and a target surface, often a sphere or a plane (see [5,6] and references

C. Barillot, D.R. Haynor, and P. Hellier (Eds.): MICCAI 2004, LNCS 3216, pp. 376–383, 2004.
© Springer-Verlag Berlin Heidelberg 2004

therein). These approaches use a well known fact from Riemannian geometry that a surface without any handles, holes or self-intersections can be mapped conformally onto the sphere, and any local portion thereof onto a disk.

In our work we focus on area preservation, motivated by a general result which ensures the existence of an area preserving mapping between two diffeomorphic surfaces of the same total area [7]. A method for building such an area preserving mapping for the purpose of cortex unfolding has been proposed in [8]. In addition, this method allows to pick, among all existing area preserving mappings, a local minimum of metric distortion. This minimum is in some sense the nearest area preserving map to a conformal one. This mathematical formulation is very promising but no numerical experiment is presented in the paper. Moreover, the numerical implementation of their method is feasible only for spherical maps.

We propose a method to evolve the cortical surface while preserving local area, and we use it to build area preserving inflated representations of the cortex. Our method relates both to [8] and [3,4,9]. We evolve a surface in time as in [3, 4,9], whereas the method of [8] is static and generates a single mapping between the cortex and a sphere. We achieve an exact area preservation as in [8], whereas in [3,4,9] area preservation is only encouraged by tangential forces. Furthermore, the latter approaches only have a discrete formulation and are specific to one type of deformable model, whereas ours is continuous and can be applied numerically both to triangulated surfaces and level sets.

To achieve this, we treat the normal and the tangential components of the motion differently. On the one hand, the normal motion controls the geometry of the surface. It is application-specific and is chosen by the user. For example, it can be a mean curvature motion in order to obtain a multiresolution representation of the cortex, or it can be a force pulling the surface towards a target shape, typically a sphere. On the other hand, given the selected normal motion, an adequate tangential motion is constructed in order to ensure an exact area preservation throughout the evolution.

The rest of the paper is organized as follows. In Sect. 2, we present a general method for building an area preserving motion for a codimension one manifold in arbitrary dimensions. Given a desired normal motion, we compute a non-trivial tangential motion such that the evolution is area preserving. In Sect. 3, we demonstrate the applicability of our method by computing area preserving inflated representations of the cortex from real brain data. Finally, we evoke some potential applications of our method in other fields in Sect. 4.

2 Area Preserving Surface Motion

In this section we first explicitely state the mathematical conditions for the preservation of the local area of an evolving codimension one manifold in arbitrary dimensions. This condition ensures that the total area, as well as the area of any patch, is preserved. When the total area needs not to be preserved, we can nonetheless preserve the relative local area, i.e. the ratio between the area of any patch and the total area. We then derive a procedure to build an area preserving or relative area preserving tangential motion. Finally we describe the numerical implementation of our method with two major types of deformable models: triangulated surfaces and level sets. In the following, we make an intensive use of differential geometry. We refer the reader to [2] for the basic theory. Note that, contrary to conformal approaches, our formulation is not limited to a genus zero surface and applies, a priori, to an arbitrary topology.

2.1 The Total, Local, and Relative Area Preserving Conditions

Let us consider an hypersurface Γ in $\mathbb{R}^n$ evolving with speed $\mathbf{v}$. We define the local areal factor J at a point of Γ as the ratio between the initial area and the current area of an infinitesimal patch around this point. So an increasing J indicates local shrinkage and a decreasing one local expansion. The preservation of total area, local area, and relative area respectively write

$$\bar{J} = 1 \ \ (1a) \quad ; \quad J = 1 \ \ (1b) \quad ; \quad J = \bar{J} \ \ (1c) \ , \tag{1}$$

where $\bar{\ }$ is the average of a quantity along Γ. $\bar{J}$ is related to the variation of the total area through $A_0 = A\,\bar{J}$. The local areal factor J of a material point evolves according to

$$\frac{DJ}{Dt} + J\,\mathrm{div}_\Gamma \mathbf{v} = 0 \ , \tag{2}$$

where D/Dt denotes the material derivative, and div_Γ denotes the intrinsic divergence operator on Γ. As a consequence, the condition to be verified by the motion for the preservation of total area, local area and relative area are respectively

$$\overline{\mathrm{div}_\Gamma \mathbf{v}} = 0 \ \ (3a) \quad ; \quad \mathrm{div}_\Gamma \mathbf{v} = 0 \ \ (3b) \quad ; \quad \mathrm{div}_\Gamma \mathbf{v} = \overline{\mathrm{div}_\Gamma \mathbf{v}} \ \ (3c) \ . \tag{3}$$

Note that the right-hand side of (3c) is spatially constant but is time-varying. Also, the preservation of local area is the combination of the preservation of total and relative area, so in the sequel we focus on (3a,c) only.

If we decompose $\mathbf{v}$ into its outward normal component v_N and its tangential part $\mathbf{v}_T$, (3a,c) become

$$\overline{H\,v_N} = 0 \ \ (4a) \quad ; \quad \mathrm{div}_\Gamma \mathbf{v}_T + (n-1)\,H\,v_N = (n-1)\,\overline{H\,v_N} \ \ (4b) \ , \tag{4}$$

where $\mathbf{N}$ is the outward normal and H is the mean curvature. In the left-hand side of (4b), we now better see the two different sources of local expansion/shrinkage: one is tangential motion, the other is the association of normal motion and curvature.

2.2 Designing a Relative Area Preserving Tangential Motion

We now outline our method to build a relative area preserving motion. We are given a normal velocity field v_N. Let us consider the solution η of the following intrinsic Poisson equation on Γ:

$$\Delta_\Gamma \eta = (n-1)\left(H\, v_N - \overline{H\, v_N}\right) \quad , \tag{5}$$

where Δ_Γ denotes the Laplace-Beltrami operator on Γ. Finding a solution of (5) is possible because the right-hand side is of zero average [10]. Moreover, the solution η is unique up to a constant. Then one can check that

$$\mathbf{v} = v_N \mathbf{N} - \nabla_\Gamma \eta \tag{6}$$

verifies (4). Note that the normal motion is not altered since $\nabla_\Gamma \eta$, the intrinsic gradient of η, is purely tangential, and that the resulting velocity is non local: it depends on the whole shape and motion of the interface.

For a given normal velocity, there are in general an infinity of admissible area preserving tangential velocities. The particular solution derived from (5) is a reasonable choice since our method outputs a null tangential motion if the normal motion is already area preserving.

In general, the given normal velocity field v_N does not preserve total area, and our method can only enforce a relative area preservation. If a strict area preservation is required, we can either apply an adequate rescaling at a post processing step, or integrate in the normal motion a rescaling motion $-\overline{H\, v_N}\,(\mathbf{x}\cdot\mathbf{N})\mathbf{N}$ so that the total area is preserved.

2.3 The Area Relaxation Term

Numerical implementations of the above method may yield deceiving results. Indeed, in pratice, (5) can only be solved up to a certain tolerance, and the surface evolution is subject to discretization errors. As a consequence, the area preserving condition cannot be fulfilled exactly and the area of interface patches may drift slowly from its expected values. Thus, area preservation is likely to be broken when the number of iterations increases.

We augment our method with an area relaxation term which encourages a uniform area redistribution whenever area preservation is lost. This feature is crucial to counteract local drifts in area due to numerical errors. We insist on the fact that this additional term is not a numerical heuristic: it is fully consistent with the mathematical formulation given above. Let us now seek the solution of

$$\Delta_\Gamma \eta = (n-1)\left(H\, v_N - \overline{H\, v_N}\right) + \lambda\left(1 - \frac{J}{\bar{J}}\right) \quad , \tag{7}$$

where λ is a weighting parameter. We build the velocity from (6) as previously. Using (2) we get that

$$\frac{D(\bar{J}/J)}{Dt} = \lambda(1 - \bar{J}/J) \quad . \tag{8}$$

Hence the ratio $\bar{J}/J$ relaxes exponentially towards unity with time constant $1/\lambda$. In our problem we have J $= \bar{J} = 1$ for $t = 0$ so that the solution of (8) is $J = \bar{J}$ for all t as desired. This was already the case without the relaxation term, so at first sight this term is of no use. But in practice, numerical errors can make $\bar{J}/J$ incidently deviate from unity. Thanks to the area relaxation term, it is now possible to recover from this numerical drift.

2.4 Numerical Methods

For each time step we have to solve the intrinsic Poisson equation (7). Apparently it represents a huge computational cost. Hopefully, only the very first iteration is expensive. Indeed, for subsequent iterations, we use the solution η of time $t-1$ as the initial guess for time t. If the shape and the normal motion change slowly relatively to the chosen time step, the guess is likely to be very close to the actual solution, and solving (7) is computationally cheap.

Triangulated surfaces: solving a Poisson equation on a triangulated surface can be done with a finite element technique as proposed in [6]. Equation (7) then translates into a linear system with a sparse symmetric positive semi-definite matrix and can be solved efficiently with numerical iterative methods such as the conjugate gradient method [11].

Level sets: at first sight, a level set implementation of cortex unfolding is not feasible. Indeed, a level set representation conveys a purely geometric description of the interface. The tangential part of the velocity vanishes in the level set evolution equation. As a consequence, it is impossible to keep point correspondences or to handle data living on the interface with the straightfoward level set approach. Some hybrid Lagrangian-Eulerian methods have been proposed to circumvent this limitation [12]. However, for sake of stability and topology independence, a completely Eulerian approach should be preferred. Recently, several Eulerian methods based on a system of coupled partial differential equations have been proposed to handle interface data in the level set framework. [13, 14] address the transport and diffusion of general data on a moving interface. In [15], we have described how to maintain explicit backward point correspondences from the current interface to the initial one. Our method consists in advecting the initial point coordinates with the same speed as the level set function. This extension of the level set method makes it applicable to cortex unfolding.

In order to solve the Poisson equation in this framework, we use a finite-difference discretization of the Laplace-Beltrami operator proposed in [16]. Equation (7) then translates into a linear system with a sparse symmetric indefinite matrix suited for a minimum residual method [11]. To compute the average of a quantity along the interface, we use a smoothed version of the Dirac function on the cartesian grid as in [17].

3 Experimental Results

In this section we focus on the level set implementation of our method and we compute inflated representations of the cortex from real brain data. Our method

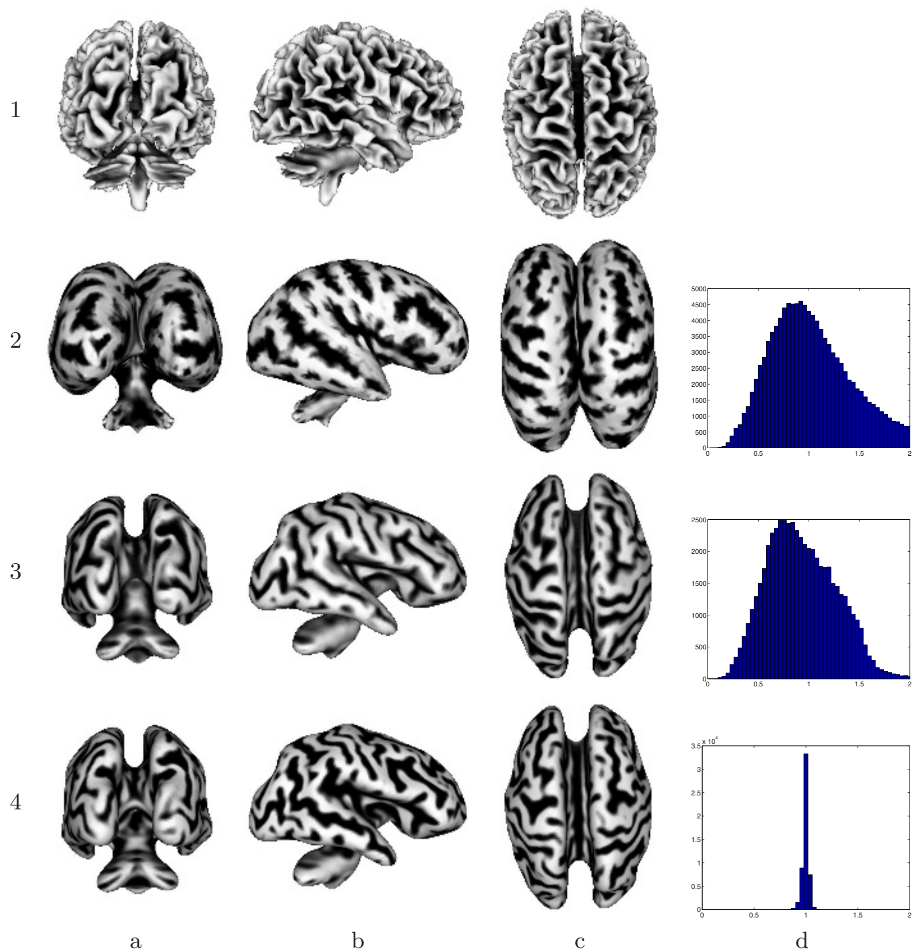

Fig. 1. Several views *(columns a,b,c)* of the initial surface *(row 1)* and of the inflated representations obtained with the popular method of Fischl et al. *(row 2)*, with a standard mean curvature motion *(row 3)* and with our method *(row 4)*. Histograms of the normalized areal factor $J/\bar{J}$ in each case *(column d)*.

is quite flexible through the choice of the normal motion, and we could obtain a sphere or any target shape by designing a force pulling the surface towards the target. In order to obtain smoothed versions of the cortex, we use the well-known and easy-to-implement motion by mean curvature $v_N = -H$ or a variant with an additional rescaling motion $v_N = -H + \overline{H^2}\,(\mathbf{x} \cdot \mathbf{N})$ in order to preserve the total area.

In the following experiments, the input of our algorithm is a $128 \times 128 \times 128$ level set of the pial surface of a human brain extracted from a $256 \times 256 \times 256$ T1 weighted MRI image with a home-made segmentation software combining hidden Markov random fields classification [18] and topology preserving level

Table 1. Standard deviation of $J/\bar{J}$ against λ when using our method.

λ	1	3	5	10
$\sigma(J/\bar{J})$	0.060	0.041	0.035	0.027

set deformable models [19]. To give an idea of the resolution of the surface, the marching cube algorithm produces roughly 100000 triangles from this level set.

In Fig. 1, in order to show the benefits of area preservation, we compare three inflated representations. The first one *(row 2)* was obtained with the nice method of Fischl et al. [1]. This method minimizes metric distortion but does not consider area. The second inflated cortex *(row 3)* was obtained with a standard mean curvature motion with a null tangential motion. The third one *(row 4)* was obtained with a mean curvature motion plus an area preserving tangential motion computed with our method. As expected, the geometries of last two representations are identical since the normal motion is the same in both cases. We display the histograms of the normalized areal factor $J/\bar{J}$ in each case *(column d)*. Not suprisingly, area distortion is far smaller when using our method. In this example, it is less that 5 percent almost everywhere. More interestingly, the overall aspect all the representations is similar, which suggests that our method does not induce a blow-up in metric distortion. Moreover, as shown in Table 1, the amount of area distortion decreases when the area relaxation parameter λ increases. On the other hand, higher values of λ require to use smaller time steps and hence to perform a larger number of iterations.

4 Conclusion and Future Work

We have presented a new method to generate unfolded area preserving representations of the cerebral cortex: depending on the normal motion driving the geometry, an adequate tangential motion is computed to ensure an exact area preservation. A demonstration of the efficiency of the method and a comparison with a popular existing method have been done on real brain data.

Future work includes implementing our method for triangulated surfaces and evaluating its benefits for the evolution of explicit deformable models in other applications. Indeed, our approach involves a good redistribution of control points which is synonymous of numerical stability: we expect it to prevent the merging of grid points or the formation of the so-called swallow tails. Thus, it could be an alternative to the heuristic elastic forces or the delooping procedures typically used to deal with these difficulties.

References

1. Fischl, B., Sereno, M., Dale, A.: Cortical surface-based analysis ii : inflation, flattening, and a surface-based coordinate system. Neuroimage **9** (1999) 195–207
2. DoCarmo, M.P.: Differential Geometry of Curves and Surfaces. Prentice-Hall (1976)

3. Dale, A.M., Sereno, M.I.: Improved localization of cortical activity by combining eeg and meg with mri cortical surface reconstruction: A linear approach. Journal of Cognitive Neuroscience **5** (1993) 162–176
4. Carman, G., Drury, H., Van Essen, D.: Computational methods for reconstructing and unfolding the cerebral cortex. Cerebral Cortex (1995)
5. Gu, X., Wang, Y., Chan, T., Thompson, P., Yau, S.T.: Genus zero surface conformal mapping and its application to brain surface mapping. In Taylor, C., Noble, J.A., eds.: Information Processing in Medical Imaging. Volume 2732 of LNCS., Springer (2003) 172–184
6. Angenent, S., Haker, S., Tannenbaum, A., Kikinis, R.: Laplace-Beltrami operator and brain surface flattening. IEEE Transactions on Medical Imaging **18** (1999) 700–711
7. Moser, J.: On the volume elements on a manifold. AMS Transactions **120** (1965) 286–294
8. Angenent, S., Haker, S., Tannenbaum, A., Kikinis, R.: On area preserving mappings of minimal distorsion. Preprint (2002)
9. Montagnat, J., Delingette, H., Scapel, N., Ayache, N.: Representation, shape, topology and evolution of deformable surfaces. Application to 3D medical image segmentation. Technical Report 3954, INRIA (2000)
10. Rauch, J.: Partial Differential Equations. Springer-Verlag, New York (1991)
11. Barret, R., Berry, M., Chan, T.F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C., van der Vonst, H.: Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods. SIAM, Philadelphia (1994) Available from netlib.
12. Hermosillo, G., Faugeras, O., Gomes, J.: Unfolding the cerebral cortex using level set methods. In Nielsen, M., Johansen, P., Olsen, O., Weickert, J., eds.: Scale-Space Theories in Computer Vision. Volume 1682 of Lecture Notes in Computer Science., Springer (1999) 58–69
13. Xu, J., Zhao, H.: An Eulerian formulation for solving partial differential equations along a moving interface. Technical Report 02-27, UCLA Computational and Applied Mathematics Reports (2002)
14. Adalsteinsson, D., Sethian, J.: Transport and diffusion of material quantities on propagating interfaces via level set methods. Journal of Computational Physics **185** (2003)
15. Pons, J.P., Hermosillo, G., Keriven, R., Faugeras, O.: How to deal with point correspondences and tangential velocities in the level set framework. In: Proceedings of the 9th International Conference on Computer Vision, Nice, France, IEEE Computer Society, IEEE Computer Society Press (2003) 894–899
16. Bertalmio, M., Cheng, L., Osher, S., Sapiro, G.: Variational problems and partial differential equations on implicit surfaces. Journal of Computational Physics **174** (2001) 759–780
17. Peng, D., Merriman, B., Osher, S., Zhao, H., Kang, M.: A PDE-based fast local level set method. Journal on Computational Physics **155** (1999) 410–438
18. Zhang, Y., Brady, M., Smith, S.: Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. IEEE Transactions on Medical Imaging **20** (2001)
19. Han, X., Xu, C., Prince, J.: A topology preserving level set method for geometric deformable models. IEEE Transactions on Pattern Analysis and Machine Intelligence **25** (2003) 755–768