A Unified Non-rigid Feature Registration Method for Brain Mapping

Haili Chui [a]

[a] *R2 Technologies, Sunnyvale, CA, USA*

Lawrence Win [b] Robert Schultz [b]

[b] *Yale Child Study Center, Yale University, New Haven, CT, USA*

James S. Duncan [c,2]

[c] *Dept. of Diagnostic Radiology, Yale University, New Haven, CT, USA*

Anand Rangarajan [*,1]

Dept. of CISE, Univ. of Florida, Gainesville, FL, USA

Abstract

This paper describes the design, implementation and results of a unified non-rigid feature registration method for the purposes of of anatomical MRI brain registration. An important characteristic of the method is its ability to take into account the spatial inter-relationships of different types of features. We demonstrate the application of the method using two different types of features: the outer cortical surface and major sulcal ribbons. Points sub-sampled from each type of feature are fused into a common 3D point-set representation. Non-rigid registration of the features is then performed using a new robust non-rigid point matching algorithm. The point matching algorithm implements an iterative joint clustering and matching (JCM) strategy which effectively reduces the computational complexity without sacrificing accuracy. We have conducted carefully designed synthetic experiments to gauge the effect of using different types of features either separately or together. A validation study examining the accuracy of non-rigid alignment of many brain structures is also presented. Finally, we present anecdotal results on the alignment of two subject MRI brain data.

Key words: Non-rigid registration, brain mapping, sulci, MR, clustering, deformation, feature-based, thin-plate splines, correspondence

1 Introduction

In human brain mapping, a spatio-temporal statistical analysis of anatomical (MRI, CT) and functional (fMRI, PET, SPECT, EEG, MEG) data across diverse modalities is required. It has become clear that the subject data from different imaging modalities (MRI, fMRI, PET etc.) have to be placed in a common spatial coordinate reference frame Toga and Mazziotta (1996) in order to facilitate a statistical analysis. One way this can be achieved is by bringing stable anatomical structures of different subjects (as seen in 3D MRI) into register. This is a daunting task since there are many anatomical features—cortical folding patterns such as sulci and gyri—that vary dramatically from subject to subject and may not always be present. Nevertheless, the existence of stable and consistent large scale anatomical structures such as the outer brain surface, major sulci and major subcortical volumes allows us to be cautiously optimistic of the success and value of 3D MRI brain registration.

There exist a plethora of 3D brain registration methods in the literature and we briefly review some of the approaches in the next section. Most approaches can be classified as either voxel-based or feature-based (with some recent methods attempting their integration). Our approach is feature-based and as with other feature-based methods, we have to first extract the anatomical features from the 3D MRI data prior to registration. In this work, we pay particular attention to the extraction of *stable, consistent and important* brain structures. Subsequently, registration is achieved by non-rigid matching of the important brain structures. Our feature registration method differs from other work in two important ways. First, we directly work on point feature locations rather than higher order features such as surface normals, curvature etc. This has the advantage of robustness since the noise in point feature locations is just point "jitter" which can be modeled. The major downside of working with point features is the well known point correspondence problem but as we shall argue, the difficulty of the correspondence problem has usually been overstated. Second, as opposed to most previous approaches, we take into account the

* Corresponding Author
Email address: anand@cise.ufl.edu (Anand Rangarajan).
URL: http://www.cise.ufl.edu/ anand (Anand Rangarajan).
[1] We acknowledge support by NSF IIS-0196457.
[2] We acknowledge support by NIH.

spatial inter-relationships between different types of point features by simultaneously aligning them in a joint homogeneous point representation space. As we will show through our experiments, the combination of different features provides important mutual anchoring information for each other and improves the registration.

We begin by briefly examining the basic elements of different current brain registration strategies.

2 Review

Most of the current efforts at inter-subject non-rigid anatomical registration can be broadly categorized into voxel-based and feature-based methods.

Voxel intensity-based approaches try to find the best deformation such that an image intensity similarity measure is maximized. Most methods in this class allow highly complex volumetric deformations to accommodate vast anatomic variations. For instance, spline models Meyer et al. (1997), elastic media models Bajcsy and Kovacic (1989); Gee (1995), viscous fluid models Christensen et al. (1997); Christensen (1999) and other local smoothness models Collins et al. (1995, 1998) are introduced as constraints to guide the non-rigid spatial mapping while maximizing a voxel intensity similarity measure.

Although the results of these methods clearly demonstrate the power of highly complex non-rigid deformations, there have been concerns as well since these methods do not attempt to match specific anatomical structures. A somewhat tacit assumption underlying voxel-based methods is that the brain structures are *matchable* as long as there is enough flexibility provided by the spatial deformation. It is already well known that minor cortical sulcal patterns may not be very consistent Toga and Mazziotta (1996), i.e., a minor sulcus in one person may not even exist in another person. By forcibly matching such non-corresponding anatomical structures, the extra flexibility of the complex volumetric deformation may make the results unpredictable and hence less reliable. To somewhat alleviate this problematic aspect, modifications have been added in some voxel-based methods to include higher level feature information. For example, landmarks were used as an initial step in Christensen et al. (1997); Joshi and Miller (2000) and major sulcal location information was incorporated via the chamfer distance measure in Collins et al. (1998). Despite these modifications, the presence of inconsistent structures across different subjects still remains an unresolved problem for most voxel-based methods.

This particular problem calls for more careful treatment of different brain structures when used for the purposes of brain registration. There are ma-

jor brain structures which are consistent across subjects and are important anatomically and/or functionally. In contrast, other minor structures may either be inconsistent, hence not matchable, or not important and hence should not be considered in the first place as they tend to increase the complexity of the registration task. By using only the features which satisfy both the consistency and the importance criteria, it not only provides us with a reliable method to handle extreme variability across subject brains, but also reduces the computational complexity. This brings us to the feature-based brain registration methods.

Obviously, features which represent important brain structures have to be extracted first. The features run the gamut of landmark points Bookstein (1989), lines, curves Davatzikos and Prince (1994); Sandor and Leahy (1997) or surfaces Thompson and Toga (1996); Thompson et al. (1997); Davatzikos (1997); Vaillant and Davatzikos (1999). After the feature extraction step, these methods then attempt to solve the resulting feature matching problem (point matching, line matching, curve matching or surface matching) for the best deformation between the features. The spatial transformations resulting from feature matching is then propagated to the whole volume. With recent advances on both the brain segmentation front using deformable models Xu et al. (1998); Zeng et al. (1999a); MacDonald et al. (2000); Wang and Staib (2000) and the feature extraction front Vaillant et al. (1996); Khaneja et al. (1998); Zeng et al. (1999b), more and more anatomical features are becoming readily available. The main question at this juncture is how to fully utilize these different types of features for brain registration. We first review some currently available methods.

In Bookstein (1989), landmarks are used for non-rigid registration and shape analysis. A thin plate spline (TPS) bending measure is minimized while ensuring that the Euclidean distance between landmarks is small. This method is now viewed as being limited by the difficulty of finding good landmark points. More recent approaches have been based on dense feature representations such as 3D surfaces. In Thompson and Toga (1996), the surfaces of the lateral ventricles and outer cortex are chosen because they are developmentally fundamental for the brain. To better represent the deep cortical structures (sulci), parametric mesh surfaces are also interactively extracted. A point-to-point mapping between each pair of the surfaces is then calculated and a linearly weighted 3D volumetric warping is generated. The work presented in Davatzikos (1997) has a similar surface-based framework. Surface curvature maps at different scales are used to model different brain structures. More consideration is given to the inhomogeneity within the brain. A more sophisticated elasticity model makes the algorithm flexible at the ventricles and powerful enough to account for some abnormal cases where, for example, tumors are involved. There has also been considerable interest in indirect surface matching approaches based on surface reparameterizations (flattening or

mapping onto a sphere).

The above feature-based methods have been successfully applied to various problems and have achieved excellent results Thompson and Toga (1996); Davatzikos (1997). However, there is still room for improvement. While each type of feature is useful by itself, these earlier methods neglected the valuable information present in the *inter-relationship* between different features by separating the features from each other during matching. More recent research efforts have been trying to capitalize on such information. For example, to improve the cortical alignment, incorporation of sulcal features into the matching framework has attracted a lot of attention. In both Thompson et al. (1997) and Vaillant and Davatzikos (1999), major corresponding sulcal curves are introduced into the mapping of the outer cortical surface. By enforcing exact correspondence of the sulcal curves within such a mapping, it has been shown that the alignment of the cortical surface can be improved. These methods work well if there is only a single surface and a few other curves, which also happen to be on that surface. It is not clear, though, how these methods can be applied to other more general situations in feature registration, where, for example, there are multiple surfaces that may or may not be connected.

We propose a general framework to attack this problem. The basic idea is quite simple and straightforward. To achieve the combination and joint registration of different types of features, we first fuse them together into a common point representation space. After this step is accomplished, non-rigid registration can be achieved by solving a point matching problem. In this work, we choose two different types of anatomical features to demonstrate the idea. The first is the smoothed outer cortical surface Zeng et al. (1999a). Smoothing of the surface is done so that the surface still closely wraps over the brain and yet all the sulci are filled. Such a surface captures the global shape of an individual brain. The second type of feature is the major sulcal ribbons Zeng et al. (1999b). Major sulci, such as the central sulcus, the sylvian fissure and the interhemispheric fissure, are chosen to bring forth more detailed shape information from within the brain. We then fuse the two different types of feature together by converting them all into a point representation, i.e., points are sub-sampled from each feature and then placed together to form a common point representation. Note that a point representation allows a unified treatment of the different features while maintaining the uniqeness of each feature type via label or attribute. The flexibility of using point representations easily overcomes the otherwise difficult problem of feature data fusion. To solve the resulting matching problem between hundreds of points, we have designed a new iterative joint point clustering and matching algorithm. Originating from our previous robust point matching (RPM) algorithm Gold et al. (1998); Rangarajan et al. (1997); Chui et al. (1999); Chui and Rangarajan (2000), the current algorithm has incorporated a few new improvements. By solving both the forward (A to B) and the reverse (B to A) deformations, it is more symmet-

ric than the previous RPM. The clustering step is carried out simultaneously with the estimation of the deformation so that the accuracy of matching is improved while reducing the computational complexity. The deformation obtained via the feature point matching process is volumetric and hence can be directly applied to the volume data without further interpolation. We now describe our approach in greater detail.

3 A Unified Feature Registration Method

The overall scheme of our method is as follows: i) choose and extract major and consistent neuroanatomical features; ii) fuse these features into a common point-set representation, iii) solve for the spatial deformation between two feature point-sets through point matching. We explain each step in detail below.

3.1 Feature Extraction

At present, we choose the outer cortical surface as well as major sulcal ribbons as the dominant features. The outer cortical surface has been widely used for brain registration since it provides a very good model for the global shape of the brain. Automated extraction of the brain surface is based on the coupled surface-based brain segmentation method as described in Zeng et al. (1999a). Since the minor sulci seen on the brain surface cannot be expected to be consistent across subjects, we decided to further smooth the cortical surface. The smoothing is done so that the surface still closely wraps over the brain but with all the sulci filled up (as shown in Figure 1). To accomplish this, we first apply Gaussian smoothing on the original MRI volume image with skull removed. The smoothing blurs all the sulci. Then we extract the smoothed outer cortical surface as an iso-surface from the blurred volume image.

The second type of feature is the set of major sulcal ribbons. Major sulci, such as the central sulcus, the sylvian fissure and the interhemisphere fissure, are chosen because of their well known importance and relative consistency across individuals. Instead of using 3D space curves, we use ribbons, which provides a better representation for the sulci's deep 3D structure (as shown in Figure 1). The interactive extraction of these ribbons is done with the help of a ribbon extraction method developed in Zeng et al. (1999b).

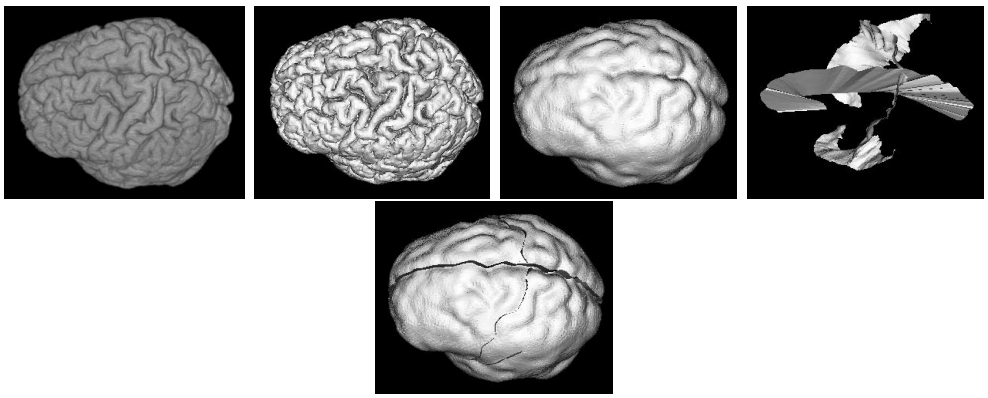

Fig. 1. **Feature extraction**. From left to right: i) original MRI brain volume; ii) extracted outer cortical surface without smoothing; iii) smoothed outer cortical surface; iv) extracted major sulcal ribbons; v) the joint feature representation with the outer cortical surface and the sulcal ribbons placed together. Most of the ribbon structures are hidden by the cortical surface.

3.2 Feature Fusion

Once all the features are extracted, we have a set of different 3D surfaces in the form of the closed outer cortical surface and open major sulcal ribbons. We then run a sampling procedure to convert each of the feature surfaces, which are parameterized as polygonal meshes, into points. All the vertices on a single feature surface are considered first. After dividing the space into cubes of equal sizes, we compute the average of all vertices lying inside each cube. The final point-set for each feature surface consists of all the average points computed in this manner. In this way, the outer cortex is reduced from the original order of 10^4(50,000 ˜ 80,000) vertices to approximately 1000 average points. The ribbon surfaces are sampled in a similar fashion except that smaller sized cubes are used to boost their relative importance since they represent much smaller structures in comparison to the outer cortical surface. Each ribbon, with originally about 1000 ˜ 2000 vertices, is eventually represented by roughly 100 ˜ 200 average points. Since all the feature surfaces involved here are relatively smooth, the sizes of the sampling cubes are chosen to achieve sufficient data reduction without losing too much shape information. Finally, the point-sets from each feature surface are combined to form a super point-set, which is used as the common point-based representation for the purposes of registration. The fusion/sampling process is demonstrated in Figure 2.

3.3 Joint Point Clustering and Matching Framework

After finishing the first step, we have a complex 3D point-set composed of hundreds of points to represent each brain. We now need to solve for an optimal

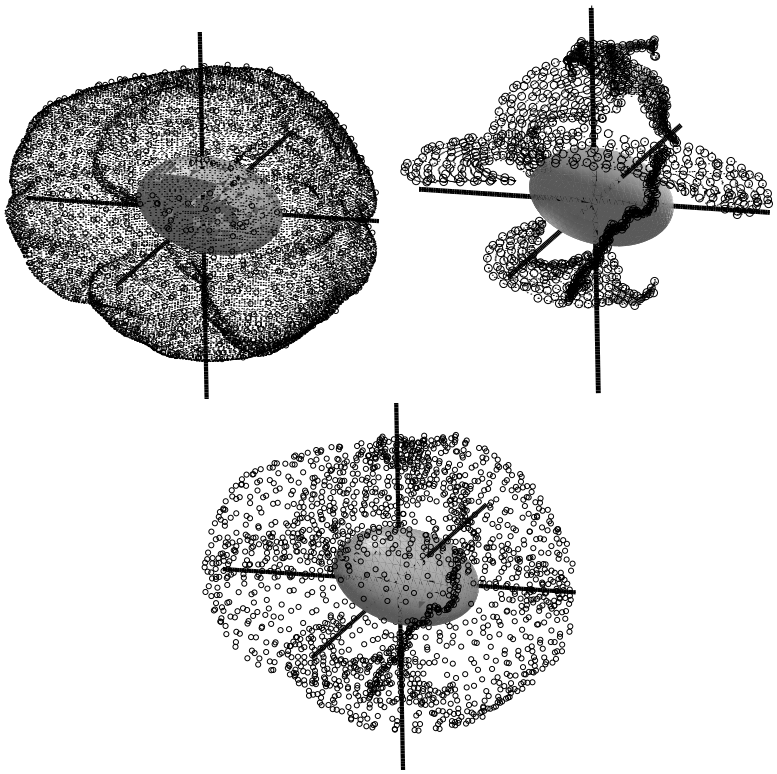

Fig. 2. **Feature fusion/sampling**. From left to right: i) sub-sampling of the outer cortical surface vertices. The original dense surface vertices are shown as dots and the sampled mean points are shown as circles. A solid ellipsoid is inserted in the middle for visualization purposes; ii) sub-sampling of the sulcal ribbons. Again, dots represent original vertices and circles represent the sampled average points; iii) placing the average points together, we form the super feature point-set.

non-rigid deformation that aligns the point-sets.

3.3.1 Joint Estimation of the Deformation and its Control Points

In order to define such deformations (usually in the form of splines), we are usually required to specify a set of *spatial control points*. The choice of the control points is vital because they directly affect the behavior of the deformation. Ideally, the control points should be chosen to accomplish the following two purposes at the same time. First, they should provide maximal necessary flexibility for the deformation to capture the variability of the data. However, too flexible a deformation is not stable so it will be easily disturbed by a small amount of noise or possible outliers. So the second requirement is that they should also provide a reasonable amount of regularization of the deformation to prevent it from becoming too flexible. These two conflicting requirements make the choice of the control point locations difficult. The situation here is actually even more difficult because we do not have much *aprior* information about the deformations. Due to this reason, predefined control points are ar-

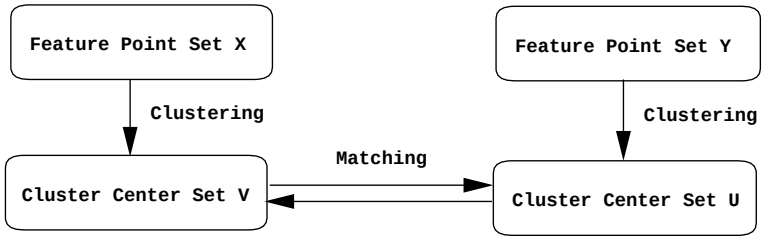

Fig. 3. The joint clustering-matching algorithm. Each point-set is clustered into a set of cluster centers (X to V, Y to U). The two sets of cluster centers are further connected through deformations that match them. Both the cluster centers and the deformation are unknown.

bitrary and problematic. Instead, we decided to include the control points as unknown variables. Subsequently, we incorporate the estimation of the control points into the overall process of estimating the deformation.

Based on the observation that the control points can better accomplish their purposes when they are placed in the more densely distributed data locations, and also the locations where the deformation is more complex, we choose to cluster the data and use the cluster centers as our control points for the deformation. Clustering is done during the process of estimating the deformation, which fulfills the requirement that there be feedback of information from the deformation estimation into the estimation of the control points.

The idea is demonstrated in Figure 3. Each of the two point-sets to be aligned is clustered to achieve a set of cluster centers, which are also used as control points to define the deformation that matches the point-sets. The cluster centers serve a dual purpose. They provide not only a concise representation of the original point data but also an optimal control point-set for the deformation. The need to specify the control points beforehand is eliminated. The only parameters now *apriori* specified are the total number of clusters and a regularization parameter.

A third and quite interesting aspect of the joint clustering and matching approach is the treatment of the correspondence problem. Since the cluster centers and deformation are jointly estimated, we can simply associate the cluster center indices of one point-set with that of the other. This defines the correspondence. With the two sets of cluster centers now evolving in lock-step, the algorithm searches for the best cluster center locations and least deformation that can match the two point-sets. A side benefit of the clustering approach is that feature consistency is automatically established. Since cluster centers and not the original point locations are matched, there is no need to have the same numbers of feature points in the two point-sets. And as mentioned previously, correspondence is established by having the same cluster center indices in the two point-sets. Hence, the resulting cluster centers can also be viewed as an automated landmarking method.

3.3.2 A Bayesian Posterior Joint Clustering and Matching (JCM) Objective Function

Here, we derive a Bayesian posterior objective function that fully expresses our joint clustering and matching goals. When viewed from a Bayesian perspective, the likelihood expresses the data clustering aspect while the prior expresses the matching aspect.

We first explain the notation. We begin with two point-sets. The reference point-set X (which we seek to warp onto a target) has N_x points. The target point-set Y has N_y points. Note that N_x can be different from N_y. We represent them as $\{x_i, i = 1, 2, \ldots, N_x\}$ and $\{y_j, j = 1, 2, \ldots, N_y\}$, while each x_i (and each y_j) represents a single point location in 3D.

A set of cluster centers (with a total of K) is associated with each point-set. The cluster centers arising from the reference point-set X is called V, consisting of K centers $\{v_a, a = 1, 2, \ldots, K\}$. The cluster centers arising from the target point-set Y is called U, consisting of corresponding K centers $\{u_a, a = 1, 2, \ldots K\}$.

A Gaussian mixture model McLachlan and Basford (1988) is used to model the clustering likelihood. Since the brain anatomical feature point-sets are highly structured, we can expect them to form tight clusters. Below, we separately write down mixture likelihoods for both feature point-sets X and Y. It should be understood that in the resulting algorithm, clustering is simultaneously performed on *both* point-sets with exactly the same number of cluster centers. Correspondence is automatic since the same index is used for the cluster centers in both point-sets. While the index is common, the locations of the cluster centers will obviously be different.

$$p(x|v, \pi^{(x)}, \Sigma) = \sum_{a=1}^{K} \pi_a^{(x)} p(x|v_a, \Sigma_a) \tag{1}$$

where

$$p(x|v_a, \sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} \Sigma_a^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x - v_a)^T \Sigma^{-1} (x - v_a)\right). \tag{2}$$

In (1) and (2), K denotes the total number of cluster centers, D the dimension of the point-sets (2D or 3D), x the vector (2D or 3D) of an instance, v_a the cluster center vector (2D or 3D), $\pi^{(x)}$ the *occupancy probability* of the clusters and Σ the cluster covariance matrix. For the sake of simplicity and ease of implementation, we immediately specialize to the case where the occupancy probabilities are uniform ($\pi_a^{(x)} = \frac{1}{K}$) and the covariance matrix Σ_a is isotropic and diagonal ($\Sigma_a = \sigma^2 I_D$). Clearly, these choices can be questioned. The

occupancy probability contains valuable information regarding the number of members in a given cluster. And, the covariance matrix gives us valuable information regarding the principal direction (tangent vector) at each cluster center. Since we are already estimating the cluster centers, the memberships of the feature points and the deformation between the two point-sets, we have elected not to excessively burden the computation by also estimating the occupancy probability and the covariance matrix at each cluster. After this specialization, the mixture likelihoods (now written for Y) become:

$$p(y|u,\sigma) = \frac{1}{K}\sum_{a=1}^{K} p(y|u_a,\sigma) \tag{3}$$

where

$$p(y|u_a,\sigma) = \frac{1}{(2\pi\sigma^2)^{\frac{D}{2}}}\exp\left(-\frac{1}{2\sigma^2}|y-u_a|^2\right). \tag{4}$$

In (3), y is a vector (2D or 3D) instance, and u_a is the ath cluster center. Note the use of a common a index and total number of clusters K for X and Y.

As mentioned previously, the prior expresses the matching aspect. Essentially, the cluster centers v and u are not independent. We relate v and u to each other via "landmark" spline deformation functions f_x and f_y. Since v_a and u_a are in correspondence, a landmark-based approach is well suited to model the deformation. However, please note that the preceding mixture likelihood clustering step can be viewed as an automated landmarking procedure since clustering is performed in lockstep on both point-sets X and Y with the cluster indices being identical as well.

$$p(v,u|f_x,f_y) = \frac{1}{Z_1}\exp\left[\sum_{a=1}^{K}\frac{1}{2\sigma^2}\left(|u_a - f_x(v_a)|^2 + |v_a - f_y(u_a)|^2\right)\right]. \tag{5}$$

In (5), there are two deformation functions f_x and f_y. The function f_x models the "forward" deformation from v to u while f_y models the "reverse" deformation from u to v. We have chosen to use two deformations to avoid being biased towards point-set X or Y. A second reason is that each function f_x actually comprises 2 (in 2D) or 3 (in 3D) separate functions (one for each coordinate of u). This separability is undesirable since it implies an unwarranted decomposition of the original 2D or 3D spline deformation into its constituent dimensions Bookstein (1989). At present, we have not attempted to constrain the reverse function f_y to be the inverse of f_x as in Christensen (1999) but we do not see any technical difficulties in imposing this constraint in the future. Finally, if a *diffeomorphism* is required, the functions f_x and f_y can be replaced by a landmark-based diffeomorphism method as in Joshi and Miller

(2000); Camion and Younes (2001). This is not technically as straightforward as imposing the inverse constraint as in Christensen (1999).

The priors on the forward and reverse functions f_x and f_y are essentially regularizers.

$$p(f_x) = \frac{1}{Z_f}\exp(-\lambda||Lf_x||^2),\ p(f_y) = \frac{1}{Z_f}\exp(-\lambda||Lf_y||^2). \tag{6}$$

In (6), L is a regularization operator. In all deformation recovery experiments, we have used the Laplacian operator which results in the familiar thin-plate spline (TPS) in 2D and in 3D Wahba (1990); Bookstein (1989). The parameter λ is a regularization parameter.

Having specified the likelihood (on X and Y) and the prior (on (u, v) and (f_x, f_y)), we may write the posterior.

$$p(v, u, f_x, f_y|x, y) = \frac{p(x|v)p(y|u)p(v, u|f_x, f_y)p(f_x)p(f_y)}{p(x, y)}. \tag{7}$$

The Bayesian MAP posterior objective function corresponding to (7) is

$$\begin{aligned} E_{\text{posterior}}(v, u, f_x, f_y) = &-\sum_{i=1}^{N_x}\log\sum_{a=1}^{K}\exp\left(-\frac{1}{2\sigma^2}|x_i - v_a|^2\right) - \sum_{j=1}^{N_y}\log\sum_{a=1}^{K}\exp\left(-\frac{1}{2\sigma^2}|y_j - u_a|^2\right) \\ &+\frac{1}{2\sigma^2}\sum_{a=1}^{K}\left(|u_a - f_x(v_a)|^2 + |v_a - f_y(u_a)|^2\right) + \lambda\left(||Lf_x||^2 + ||Lf_y||^2\right) \end{aligned} \tag{8}$$

Equation (8) is the joint clustering and matching objective function.

As it stands, minimizing (8) is awkward due to the $\sum\log\sum\exp$ forms appearing in the objective. This is a well known problem in Gaussian mixture modeling Redner and Walker (1984). The well known expectation-maximization (EM) algorithm McLachlan and Basford (1988) can be pressed into service as an optimization algorithm. Since the mixture likelihood is non-convex, the EM algorithm is usually executed many times with varying random initial conditions. We instead adopt a deterministic annealing approach Rose et al. (1990); Yuille et al. (1994); Hofmann and Buhmann (1997).

The main difference between the traditional EM algorithm for mixtures and a deterministic annealing algorithm is in the treatment of the isotropic variance parameter σ^2. In deterministic annealing, the variance parameter is imposed from without rather than being estimated from within. A temperature parameter $T = 2\sigma^2$ as in simulated annealing (or MCMC) is gradually lowered

from high values to low values. When the temperature T is high, the cluster centers congregate around the center of mass of the point-sets. As the temperature is lowered, a series of symmetry-breaking "phase transitions" Rose et al. (1990) occur during which the cluster centers progressively move away from the center of mass and toward their more local members.

Consider the following Gaussian mixture likelihood objective function:

$$E_{\text{mix}}(v) = -\sum_{i=1}^{N_x} \log \sum_{a=1}^{K} \exp\left(-\frac{1}{2\sigma^2}|x_i - v_a|^2\right). \tag{9}$$

The objective function in (9) is a straightforward mixture objective unencumbered by the deformation prior. Now consider

$$E_{\text{cmp}}(v, m^x) = \sum_{i=1}^{N_x}\sum_{a=1}^{K} m_{ai}^x |x_i - v_a|^2 + T\sum_{i=1}^{N_x}\sum_{a=1}^{K} m_{ai}^x \log m_{ai}^x \tag{10}$$

The objective function in (10) has a new variable m^x and the temperature parameter T. It turns out that

$$\min_{m^x} E_{\text{cmp}}(v, m^x) = E_{\text{mix}}(v) \tag{11}$$

when m^x satisfies $m_{ai}^x > 0$ and $\sum_{a=1}^{K} m_{ai}^x = 1$ Hathaway (1986); Yuille et al. (1994) and when T is identified with $2\sigma^2$. The new variable m_{ai}^x is a *membership* variable indicating the degree to which each point feature x_i belongs to cluster center v_a. The main convenience resulting from using (10) rather than (9) is that (10) does not have the $\sum \log \sum \exp$ form in it. Also, the term $\sum_{ai} m_{ai}^x \log m_{ai}^x$ is an *entropy* barrier function with T being the temperature. We now perform this conversion from the mixture objective in (8) to a new deterministic annealing objective along the same lines as the transition from (9) to (10).

$$\begin{aligned} E_{\text{cmp}}(v, u, f_x, f_y, m^x, m^y) = &\sum_{i=1}^{N_x}\sum_{a=1}^{K} m_{ai}^x |x_i - v_a|^2 + \sum_{j=1}^{N_y}\sum_{a=1}^{K} m_{aj}^y |y_j - u_a|^2 \\ &+ \sum_{a=1}^{K} |u_a - f_x(v_a)|^2 + \sum_{a=1}^{K} |v_a - f_y(u_a)|^2 + \lambda T||Lf_x||^2 + \lambda T||Lf_y||^2 \\ &+ T\sum_{i=1}^{N_x}\sum_{a=1}^{K} m_{ai}^x \log m_{ai}^x + T\sum_{i=1}^{N_x}\sum_{a=1}^{K} m_{aj}^y \log m_{aj}^y \end{aligned} \tag{12}$$

where $m_{ai}^x \in [0,1]$ and $m_{aj}^y \in [0,1]$ satisfy the constraints:

$$\sum_{a=1}^{K} m_{ai}^x = 1, \text{ for } i = 1, 2, \ldots N_x, \tag{13}$$

and

$$\sum_{a=1}^{K} m_{ai}^y = 1, \text{ for } j = 1, 2, \ldots N_y. \tag{14}$$

The objective function in Equation 12 is closely related to the objective functions used in our previous work on robust point matching. More detailed explanations are available in Gold et al. (1998); Rangarajan et al. (1997); Chui et al. (1999); Chui and Rangarajan (2000) showing that point matching can also be viewed as a fuzzy assignment optimization problem. Here we intend to give some brief but more intuitive explanations for all the terms used within the above objective function.

The first two terms are average residue distance measures between the data and the cluster centers. Note that the memberships present in the distance measure are themselves unknown. These two terms basically measure the degree of fidelity of the cluster centers (V and U) to the data (X and Y) respectively.

The next two terms in the objective function in (12) try to find the best deformation (both forward and reverse at the same time) to match the two sets of cluster centers. Instead of matching the original data points, the deformation estimation step attempts to match the cluster centers. Since the cluster centers are actually the control points for the deformation splines, the deformation is directly affected by a change in the cluster centers. This feedback loop between the deformation estimation step and the clustering step allows us to more closely model the deformation between the two data-sets. Such a feedback loop is only possible when we allow the cluster centers (control points) to be dynamically estimated while the deformation is being updated.

The fifth and the sixth terms play the role of regularization. The parameter λ is a weight parameter which controls the degree of deformation; larger the regularization parameter, smaller the extent of deformation and vice-versa. Note that the temperature parameter now modulates the regularization. At high temperatures, the regularization is very large, thereby forcing the deformation to be close to the identity mapping. As the temperature is lowered, the cluster centers move away from the centroid of the two point-sets and the deformation moves away from the identity mapping. This simultaneous movement is governed by the temperature parameter setting.

Before we explain the last two terms, we first discuss some interesting properties of membership variables. Our clustering membership variables m^x and m^y are continuous variables in the interval $[0, 1]$, which still satisfy the constraints that the total membership of each data point in all clusters is one. The continuous value of the membership variable reflect the "fuzziness" in our clustering model. For example, if all m^x_{ai} are the same, the membership of a data point in a cluster center is uncertain. A small calculation shows that this effectively causes all the cluster centers to lie at the center of mass of the data point-set. At the other end of the spectrum, if all m^x_{ai} are close to binary values (either 0 or 1), each cluster center will represent a separate subset of the data ponits resulting in a good representation of the shape of the data point-set. Between these two extremes, the membership matrix m^x_{ai} of some intermediate fuzziness would then generate a set of cluster centers which can capture the shape of the data points at some intermediate level. The shapes of the intermediate levels are very helpful in our quest for a good non-rigid deformation. They are much simpler than the actual data shape, which make them easier and more stable to match. On the other hand, they resemble the actual data shape to some extent. So the answer for the deformation found at a less detailed level can be used as a good initialization to find the deformation at a more detailed level. If we have a way to gradually reduce the fuzziness in a controlled manner while progressively improving our estimation of the deformation, it obviously leads to a coarse-to-fine, scale-space like strategy.

Deterministic annealing accomplishes this purpose by adding the entropy terms $T \sum_{ai} m^x_{ai} \log m^x_{ai}$ and $T \sum_{ai} m^y_{ai} \log m^y_{ai}$ to the original energy function. The temperature parameter T now controls the fuzziness of the membership matrices: higher the temperature, greater the fuzziness. As mentioned previously, this form of the entropy term effectively leads to Gaussian clusters. The square root of T can then also be regarded as the size of the clusters. The required fine control of the fuzziness can be achieved by gradually reducing T.

3.3.3 The Joint Clustering-Matching Algorithm

The resulting joint clustering-matching algorithm is very simple. It essentially involves a dual update process embedded within an annealing scheme. The update of each variable is calculated basically by differentiating the energy function w.r.t. that variable and setting the result to zero. We first briefly describe the two update steps.

Step 1. Clustering: Update membership matrices and cluster centers.

$$m^x_{ai} = \frac{q^x_{ai}}{\sum_{a=1}^{K} q^x_{ai}}, \; m^y_{aj} = \frac{q^y_{aj}}{\sum_{a=1}^{K} q^y_{aj}}, \tag{15}$$

where,

$$q_{ai}^{x} = e^{-\frac{|x_i - v_a|^2}{T}},\ q_{aj}^{y} = e^{-\frac{|y_j - u_a|^2}{T}}, \tag{16}$$

and then,

$$v_a = \sum_{i=1}^{N_x} \frac{m_{ai}^{x} x_i}{2} + \frac{f_x(u_a)}{2},\ u_a = \sum_{j=1}^{N_y} \frac{m_{aj}^{y} y_j}{2} + \frac{f_y(v_a)}{2}. \tag{17}$$

Step 2. Matching: Update the deformation functions. This is a standard least-squares spline fitting problem.

$$f_x = \arg\min_{f_x} \left(\sum_{a=1}^{K} |u_a - f_x(v_a)|^2 + \lambda ||L f_x||^2 \right), \tag{18}$$

$$f_y = \arg\min_{f_y} \left(\sum_{a=1}^{K} |v_a - f_y(u_a)|^2 + \lambda ||L f_y||^2 \right). \tag{19}$$

The update presented for the cluster centers has been slightly simplified. The deformations are held fixed when the cluster centers are updated despite the fact that the deformation is actually a function of the cluster centers. Including the complete relation into the formulation greatly complicates the calculation. Another reason for this simplification is that we are only slowly refining the deformations during this whole iterative process; it is reasonable to assume that deformations estimated at successive iterations will not be vastly different. The approximation based on the previous iteration should be most likely sufficient.

Annealing: As before Gold et al. (1998); Rangarajan et al. (1997); Chui and Rangarajan (2000), an annealing scheme (for the temperature parameter T) controls the dual update process. Starting with a high value T_{init}, the temperature parameter T is gradually reduced according to a geometric annealing schedule, $T_{new} = T_{old} \cdot r$ (r is called the annealing rate). The dual update is repeated until convergence at each temperature. The temperature T is then lowered and the process is repeated until the final temperature T_{final} is reached.

The parameter T_{init} is set to the largest square distance of all point pairs. We set r to be 0.97 (normally between $[0.9, 0.99]$) so that the annealing process is slow enough for the algorithm to be robust, and yet not too slow. To prevent overfitting, the parameter T_{final} should be set according to the amount of noise within the data set. In this work, we make a simplified treatment to set T_{final} to be equal to the average of the squared distance between the nearest

neighbors within the set of cluster centers which are being deformed. The interpretation is that at T_{final}, the Gaussian clusters for all the points will then barely overlap with each other.

Joint Clustering-Matching (JCM) Algorithm Pseudo-Code:

Initialize T, f_x and f_y.

Dual Update:

· Clustering step: update $\{m_{ai}^x, m_{ai}^y, v_a \text{ and } u_a\}$.
· Matching step: update f_x and f_y.

Anneal: $T = T \cdot r$ until T_{final} is reached.

3.3.4 Choice of Splines to Model the Deformation

We now specify the deformation parameterization in order to complete the algorithm specification. Although it is almost impossible to specify the exact deformation that can account for all of the inter-subject brain structural differences, many types of deformation models originating from either continuum mechanics or spline theory have been used to provide reasonable approximations. Splines have been especially popular for feature-based methods Thompson and Toga (1996); Davatzikos (1997) because of their well established geometrical properties. However, when it comes to the specific choice of the spline, there are few guidelines as to which spline is the most suitable. Since our framework is general and can accomodate different splines, we are able to conduct controlled comparative studies between different splines.

We implemented two types of radial basis function splines Wahba (1990). Given a set of control points $\{v_a, a = 1, 2, \ldots n\}$, a radial basis function basically defines a spatial mapping which maps any location x in space to a new location $f(x)$, represented by,

$$f(x) = \sum_{a=1}^{n} c_a \phi(|x - v_a|) \tag{20}$$

where $|\cdot|$ denotes the usual Euclidean norm in 3D and $\{c_a\}$ is a set of mapping coefficients. The kernel function ϕ assume different forms. If we choose $\phi(r) = exp(-r^2/\sigma^2)$, it becomes a Gaussian Radial Basis Function (GRBF). The parameter σ controls the locality of each kernel function. A small value of σ generates more localized and hence less smooth warpings. A different choice of using $\phi(x) = -r$ leads to another type of radial basis function called the Thin Plate Spline (TPS). Compared to the GRBF, TPS has a more global

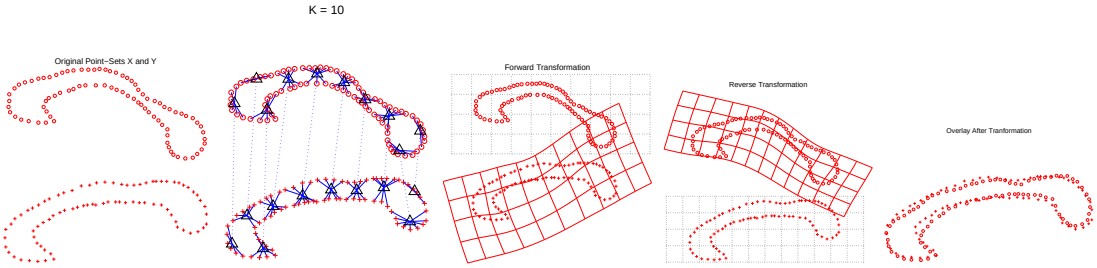

Fig. 4. **Corpus callosum example:** $K = 10$. Leftmost: original point-sets. Left of center: Cluster centers and correspondence. Center: Forward transformation. Right of center: Reverse transformation. Rightmost: Overlay of point-sets

nature—a small perturbation of one of the control points always affects the coefficients corresponding to all the other points as well. It is worth pointing out that TPS has one less free parameter. Another nice property of the TPS is that it allows the deformation to be cleanly decomposed into a rigid (affine) and a non-rigid component.

Due to space considerations, we have not included here the detailed solutions to the spline fitting problem (Equation 18 and 19). We would like to just point out that there are closed form analytic solutions available for both GRBF and TPS Wahba (1990) since they are essentially linear least-squares problems. Though derived from geometrical features, both deformations are defined over the whole 3D space and can hence be directly applied to the volume data.

4 Experiments and Results

We begin by illustrating the way the JCM algorithm works on 2D point-sets. We use two typical 2D corpus callosum shapes for the illustration.

A crucially important parameter in JCM is the number of cluster centers K. Below, we show the results of executing the JCM algorithm with $K = 10$ through $K = 60$ cluster centers. The 2D shape has about 90 points. Figures 4-9 take us through this sequence of increasingly better approximations of the shape. It is clear from the figures that the corpus callosum shape is well represented after $K = 30$. Until that point, the shape is pinched in the middle (as seen from the cluster center locations).

We are also interested in gauging the performance of the JCM algorithm on candidate point-sets. A 2D template was first chosen. Both noise and deformation performance were studied. Instead of TPS, we use a different non-rigid mapping, namely Gaussian radial basis functions (RBF) Yuille and Grzywacz (1989) for the random transformation. The coefficients of the RBF were sampled from a Gaussian distribution with a zero mean and a standard deviation

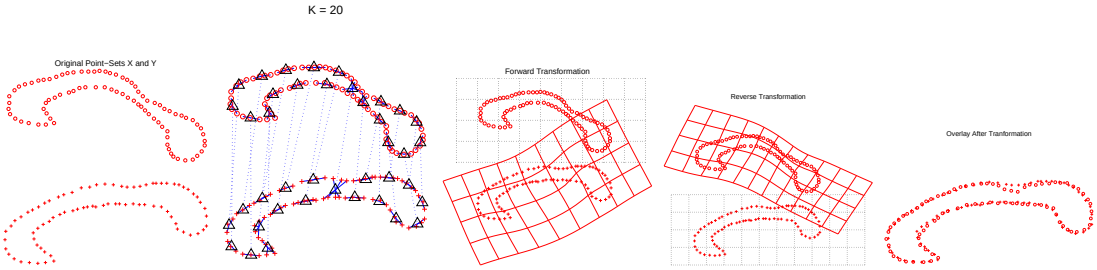

Fig. 5. **Corpus callosum example:** $K = 20$.

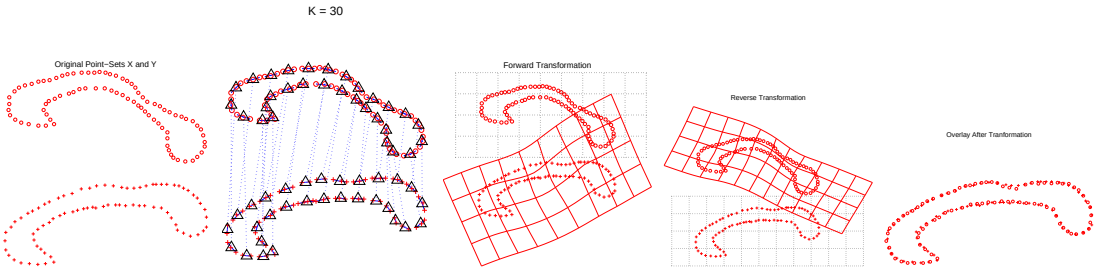

Fig. 6. **Corpus callosum example:** $K = 30$.

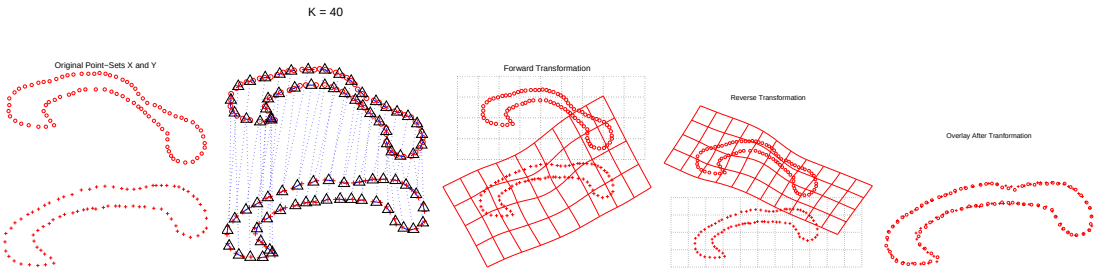

Fig. 7. **Corpus callosum example:** $K = 40$.

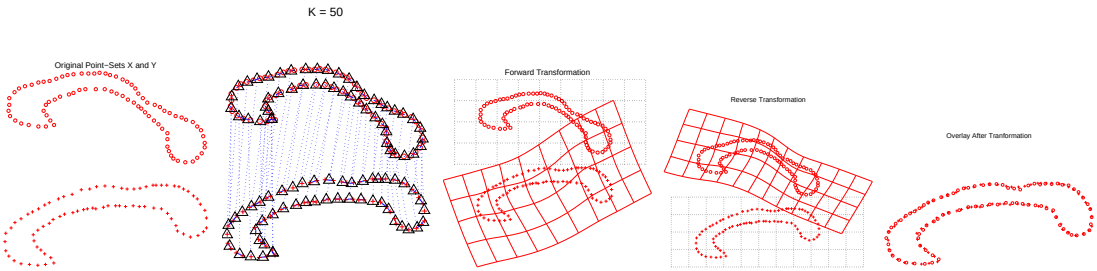

Fig. 8. **Corpus callosum example:** $K = 50$.

s_1. Increasing the value of s_1 generates more widely distributed RBF coefficients and hence leads to generally larger deformation. Random noise are added to the warped template to generate the target point-set. We then used JCM find the best TPS to map the template set onto the target set. The errors are computed as the mean squared distance between the warped template using the TPS found by the algorithms and the warped template using the ground truth Gaussian RBF.

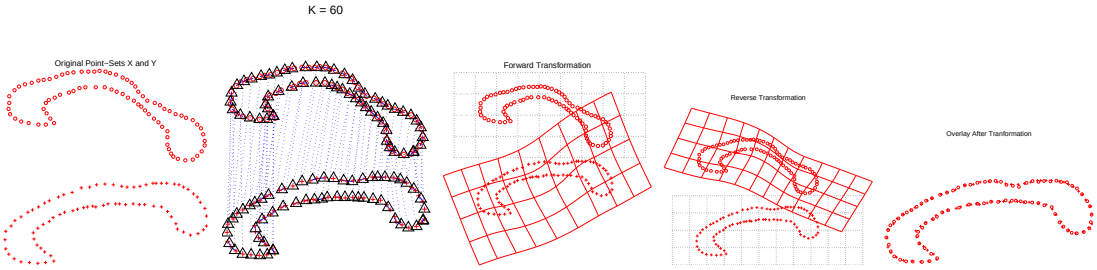

Fig. 9. **Corpus callosum example:** $K = 60$.

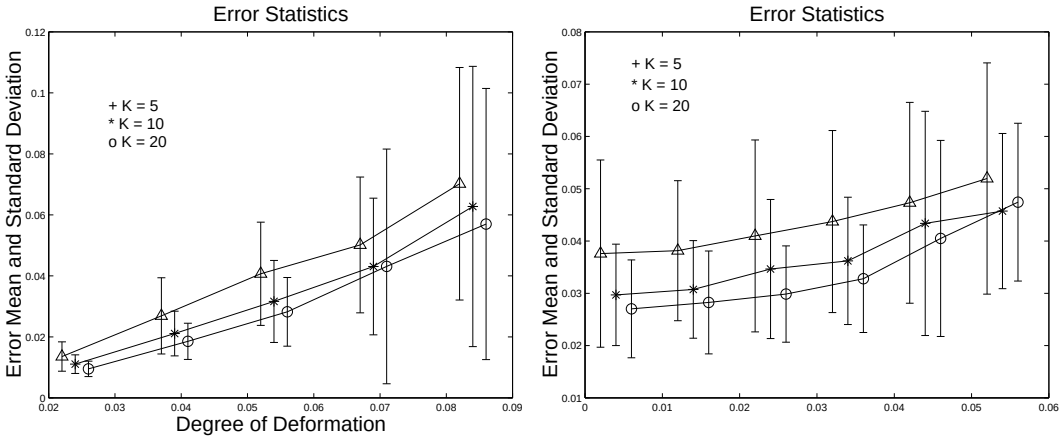

Fig. 10. Left: Deformation and Right: Noise performance on 100 trials

We conducted two series of experiments. In the first series of experiments, the template was warped through progressively larger degrees of non-rigid warping. The warped templates were used as the target data without adding noise or outliers. The purpose is to test the algorithms' performance on solving different degrees of deformations. In the second series, different amounts of Gaussian noise (standard deviation s_2 from 0 to 0.05) were added to the warped template to get the target data. A medium degree of warping was used to warp the template. The purpose is to test the algorithms' tolerance of noise. 100 random experiments were repeated for each setting within each series. The results are shown in Figure 10. Surprisingly, we did not observe much variation as K was increased. We speculate that the performance is likely to show quantized jumps as K is increased corresponding to the cluster center phase transitions. Clearly, this needs to be studied in greater detail.

We conducted various experiments on both synthetic and real data to evaluate our algorithm. Experiments on real data is clearly necessary since that is the ultimate goal of the algorithm. However, because the ground truth for real data in inter-subject brain registration is usually not available, the synthetic data provides a good alternative for validation purposes. In our experiments with synthetic data, we intend to answer the following questions: i) does the fusion of different types of features improve the registration or not? ii) if it does, what is the degree of improvement? With these questions in mind, we

first describe the synthetic experiments and the validation procedure.

4.1 Experiments on Synthetic Data

4.1.1 The Design of the Synthetic Experiment

The synthetic experiments for registration are normally carried out in the following steps: i) construct a template; ii) construct a target from the template via a synthetic deformation (ground truth); iii) recover a good deformation (via the algorithm) to match the template to the target; iv) examine the errors between the solution and the ground truth deformation.

In order to make the synthetic study more reliable, we carefully designed the experiment to eliminate any bias due to the fact we have knowledge of the ground trouth deformation. Essentially, knowledge of the ground truth deformation should not affect the algorithm's performance. For example, if one type of spline is used to construct the synthetic target data, a different spline should be used for subsequent recovery. Similar rules apply for the examination of the errors as well. Instead of using the features directly involved in the matching to evaluate the matching error, different features or the volume data should be used. Keeping these considerations in mind, we briefly describe the design of each step for the synthetic study.

4.1.1.1 Construction of the Template: We choose one normal male brain MRI (without the skull) as the raw data for our template. For the purposes of registration, as explained before, the smoothed outer cortical surface and a set of major sulcal ribbons are extracted as features.

We need to prepare the template for later error measurement as well. Apart from these features used for the registration, we asked a neuroanatomy expert to extract a different set of landmark points over the whole brain volume to get a rough error measurement. The landmark points includes two sub-groups: one group distributed on the outer cortical surface and another group distributed at critical locations of the sub-cortical structures (as shown in Figure 11). The idea is that while the error calculated over the whole landmark point-set will give us a rough global estimate of how good the alignment is, the errors from each sub-group can tell us a little bit more about where the error originates.

To provide even more detailed measurement than landmarks, we also have the MRI volume fully segmented and labeled with the help of neuroanatomy expert. A label is assigned to each voxel according to which structure the voxel belongs. A hierarchy of structures is used. The volume is first segmented into

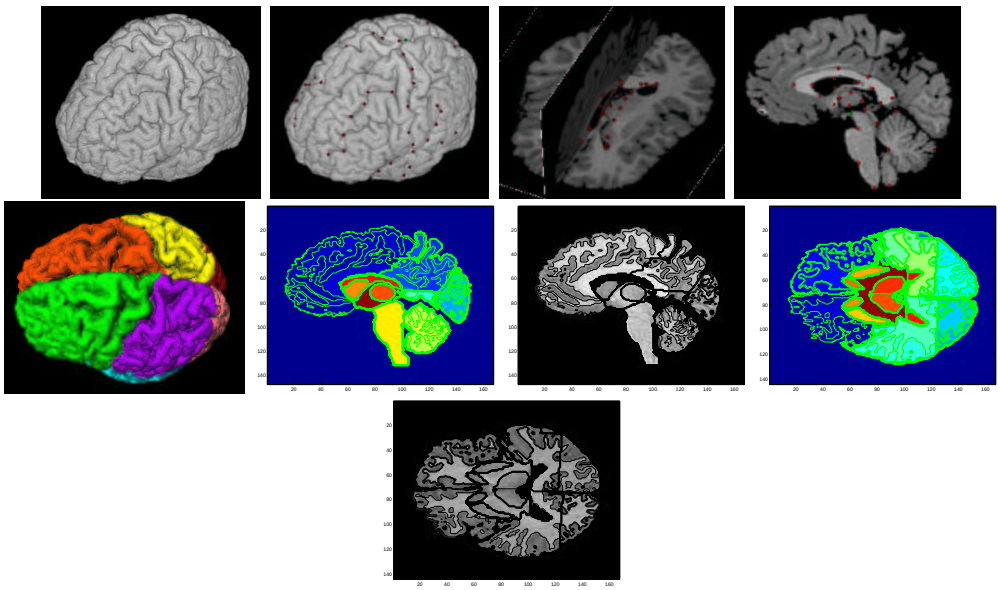

Fig. 11. Template for the Synthetic Study. Top row, from left to right: i) the template MRI volume; ii) the cortical landmarks; iii) the subcortical landmarks in 3D view; iv) again the sub-cortical landmarks in 2D view.
Bottom row, from left to right: i) a surface rendering of the fully segmented brain MRI volume; ii) a 2D slice of the segmentation. Different colors stand for different labels; iii) the corresponding gray level MRI slice with the contours from the segmentation; iv, v) another slice.

the background and the brain. The brain is then divided into cortex and sub-cortex. The cortex is subdivided into different lobes with gray/white matter segmentation performed for each lobe. Within the sub-cortex volume, a list of important sub-cortical structures are segmented including the thalamus, caudate, putamen, brain stem and the ventricles. With this finely segmented brain volume template, we can then make very detailed error measurements.

4.1.1.2 Using GRBF as Synthetic Deformation to Construct the Target: We choose GRBF as the synthetic deformation to warp the template data. The main reason is that with its locality parameter σ, GRBF can easily generate both local and global warpings. By comparing the algorithm's performance under these different circumstances, it provides another valuable way to evaluate the algorithm's performance. The control points for GRBF are manually chosen to be a group of grid points (64 points total) over the whole volume.

4.1.1.3 Using TPS to Recover the Deformation: We choose TPS as the deformation spline in our feature registration/point clustering-matching algorithm. Using TPS instead of GRBF obviously makes the problem more

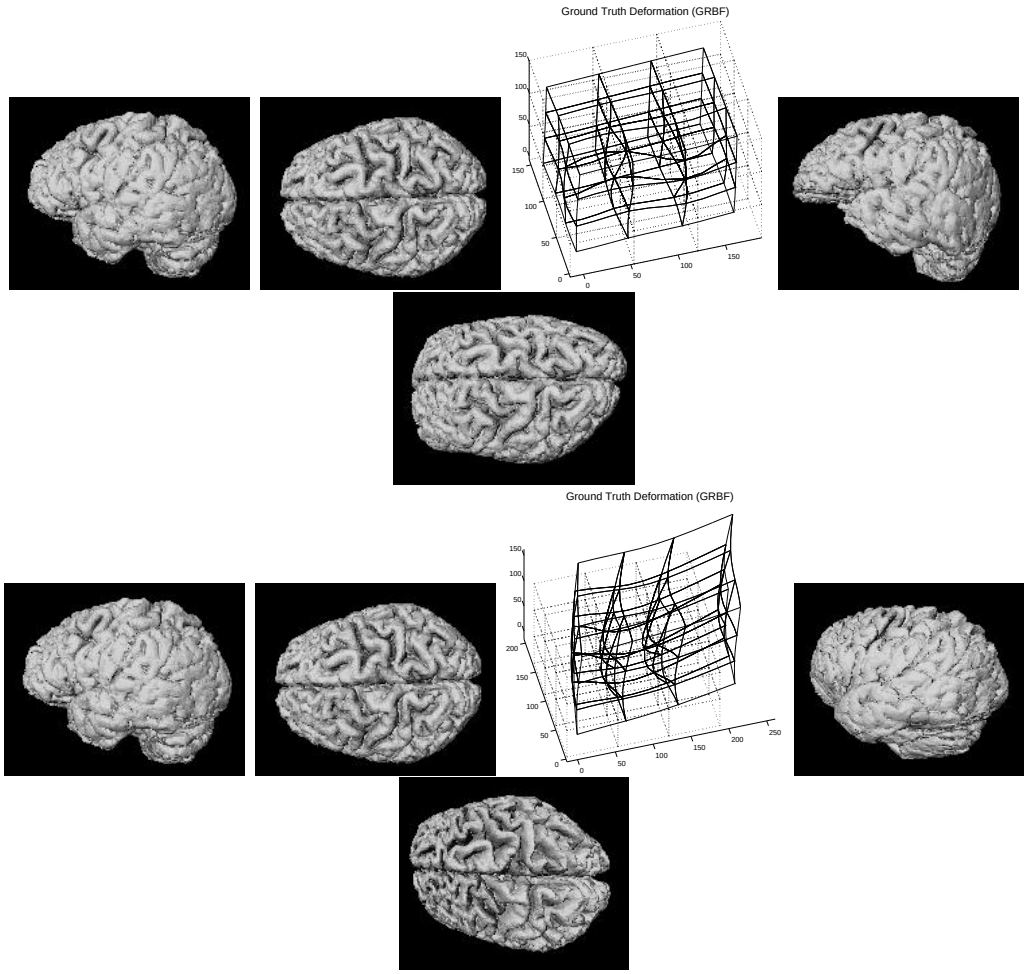

Fig. 12. Ground Truth GRBF (Local Warp and Global Warp). Top row, from left to right: i, ii) two side views of the original brain volume; iii) a randomly generated GRBF with a small $\sigma = 30$ (local warp). The original space is shown by the regular dotted 3D grid and the warped space by the solid deformed grid; iv, v) two side views of the warped brain volume by GRBF.
Bottom row, from left to right: i, ii) original brain volume; iii) a GRBF with a big $\sigma = 60$ (global) warp. Note the difference between the local warp and the global warp; iv, v) the warped volume.

difficult. However, it is necessary since it provides more unbiased validation of the algorithm. The advantage of using TPS is the smaller number of free parameters compared to GRBF. Apart from the annealing parameters, the only extra parameter that TPS needs is the regularizationparameter λ. The value for λ is manually chosen. The annealing parameters are set as discussed above. We specify the total number of cluster centers K to be 150 to give TPS enough flexibility.

To answer the question posed above, we run the JCM algorithm with different settings to see if the combination of features really improves the registration. Three different choices of features are compared— the outer surface alone,

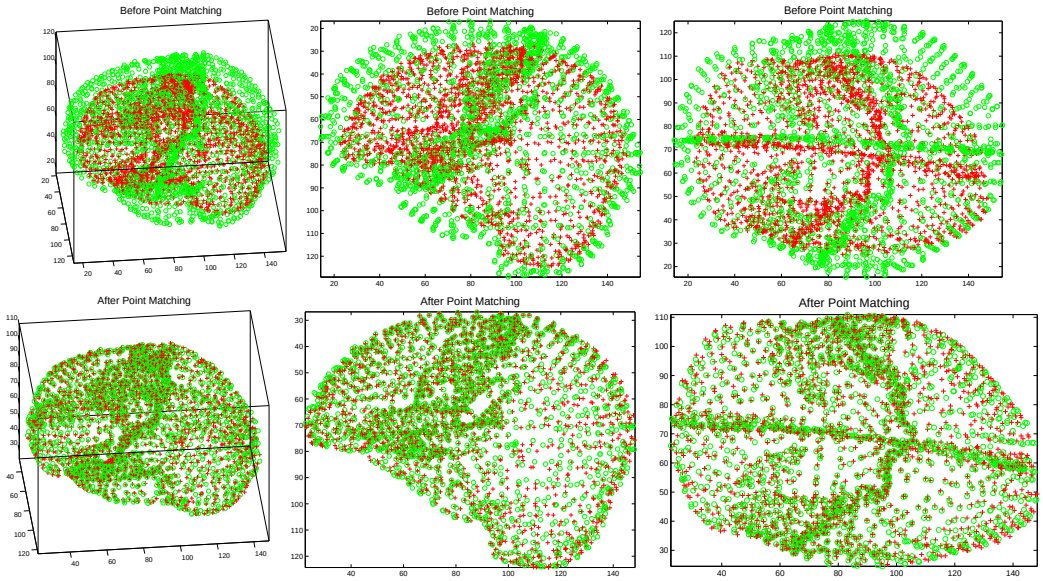

Fig. 13. Joint point clustering-matching using TPS. Top row, from left to right: i) a 3D view of 2 feature point-sets before point matching. One point-set is shown using crosses and another using circles; ii, iii) two 2D views.
Bottom row, from left to right: i) a 3D view of the feature point-sets after point matching. TPS is used to deform one point-set (circles) to match the other set; ii, iii) two 2D views.

the sulcal ribbons alone and the combination the outer surface and the sulcal ribbons. One matching example of using both the outer surface and the sulcal ribbons is demonstrated in Figure 13 and 14.

4.1.1.4 Examination of the Errors: Errors are calculated based on both the landmarks as well as the labeled volume. To measure the errors on landmarks, we first warp the original landmarks with the ground truth GRBF to get the ground truth set A. After the TPS registration, we warp the original landmarks again with the TPS recovered by the algorithm to get the solution set B. Errors are then calculated between A and B as the Euclidean distances between the two warped landmark point-sets. A similar procedure is performed for the labeled volume as well. The only difference between the landmark error and the volume error measurements is that in the latter, the error is measured as the ratio of misaligned voxels between the two warped labeled volumes for a certain segmented structure. First the misaligned voxels are counted for a particular structure. The error ratio is calculated by dividing the misaligned count by the actual total number of voxels in that structure. Such a error ratio basically provides a relative measurement for the alignment of each structure.

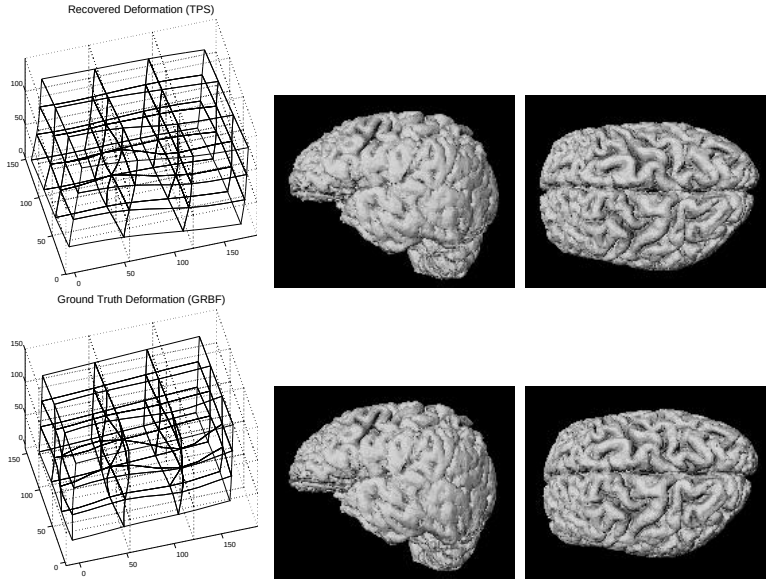

Fig. 14. Comparison of TPS and GRBF. Top row: i) TPS deformation recovered from the point matching; ii, iii) the warped brain volume using TPS.
Bottom row: i) the ground truth GRBF deformation; ii, iii) the warped brain with the GRBF. Note that though the warped volumes from TPS and GRBF look almost the same on the surface, the two deformations are still slightly different over the whole space. It shows that the two splines have different behaviors.

4.1.2 Synthetic Experiments and Results

We carried out 2 series of synthetic experiments: one with a smaller value of $\sigma = 30$ in GRBF for more localized warpings and one with a larger value of $\sigma = 60$ for global warping. Ten randomly generated trials (with randomly generated GRBF coefficients) are included in each series. The algorithm is run 3 times for each trial, each time with a different choice of feature—outer surface alone (method I), sulcal ribbons alone (method II) and the combination of outer surface and sulcal ribbons (method III). The errors for each method is averaged over the total 10 trials to get both the mean and the standard deviation. The error statistics are shown in Figure 15 and 17.

From the error statistics based on landmarks (Figure 15), method III is clearly shown to be superior than the other two methods. It brings the landmarks within 1 ˜ 2 voxels' distances consistently despite the fact that we are trying to use TPS to match to a target which is generated by a different spline (GRBF). The results confirm that the combination of the two types of features does improve the registration.

The data also reveal other interesting facts. Since all our features are mostly located at the cortical regions, it is not surprising that the alignment of the cortical landmarks tends to be much better than the sub-cortical part. Includ-

(Mean/Std., Unit: Voxel)	Method I	Method II	Method III
All Landmarks	2.83/0.61	1.96/0.29	1.58/0.30
Cortical Landmarks	2.23/0.48	1.88/0.29	1.15/0.30
Sub-Cortical Landmarks	3.46/0.97	2.05/0.46	2.03/0.38

Fig. 15. The error statistics based on the landmarks of test series 1 (local GRBF warp). Method I: using outer cortical surface alone. Method II: using sulcal ribbons alone. Method: using both together. Overall, method III gives smallest errors. Also note that the cortical landmark errors tend to be smaller than the sub-cortical landmark errors.

(Mean/Std., Unit: Voxel)	Method I	Method II	Method III
All Landmarks	2.17/0.82	1.84/0.32	1.23/0.26
Cortical Landmarks	1.58/0.58	1.57/0.30	0.83/0.25
Sub-Cortical Landmarks	2.79/1.00	2.12/0.48	1.88/0.41

Fig. 16. The error statistics based on the landmarks of test series 2 (global GRBF warp). Note that the errors generally improve a little bit when compared to test series 1.

ing further features to represent the sub-cortical structures certainly should help. One interesting fact is that method II with the sulcal ribbons actually outperforms method I, which uses the outer cortical surface in places. This is not what we originally expected. Even though the major sulcal ribbons are extended more into the brain, they seem to be too sparse a representation. The results from the experiments clearly indicate their better 3D placement outweigh their major disadvantage, namely sparseness. The ellipsoidal shape of the outer cortical surface is also more likely to cause rotation errors. Generally the behavior of TPS is more global when compared to GRBF. So it is natural that TPS approximates GRBF better when the warping is more global. This is confirmed by smaller errors from test series 2.

As shown in Figure 17, the error statistics measured based on the segmented volume data not only confirm all our findings based on the landmarks but also provide more detailed information. While its errors on the cortical lobe regions are comparable to other methods, method I (with only the outer cortical surface) simply cannot provide enough anchoring information when it comes to the sub-cortical structures. On the other hand, method II (with only the sulcal ribbons) clearly suffers from the sparseness of its feature representation. For any structure that is relatively far away from the sulcal ribbons (e.g. cerebellum) , method II leads to large errors. All these problems can be avoided by combining them together as in method III.

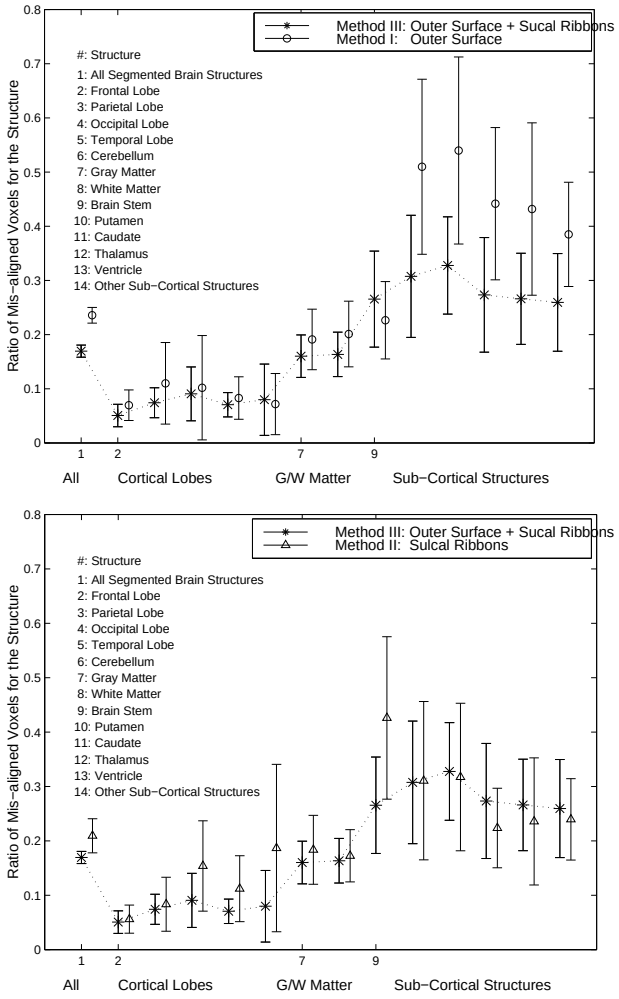

Fig. 17. The error statistics based on the volume of test series 1 (local GRBF warp). Left, comparison of method III and I. Right, comparison of method III and II. Method I, with only the outer surface, yields much bigger errors for all sub-cortical structures. Method II, with only the sulcal ribbons, tend to perform worse near the rear region of the brain (occipital lobe, parietal lobe, cerebellum and brain stem), from where all the ribbons are relatively far away. Note: when measuring errors on the brain lobes, the distinction between the gray matter and the white matter is neglected to provide a more global and overall evaluation of the alignment.

The inclusion of error examination on gray and white matter allows us to check the alignment at a much detailed level. For very convoluted structures like the gray/white matter interface, small local misalignments can accumulate to a large error. Compared to the relatively low errors on lobe alignment, the residual errors at the gray/white matter interface are much larger. This indicates that the alignment at a fine local level can be improved. The small difference between TPS and GRBF is magnified and clearly demonstrated here. The volume error statistics of test series 2 is similar to test series 1 but with a slight improvement and hence, we do not repeat them here.

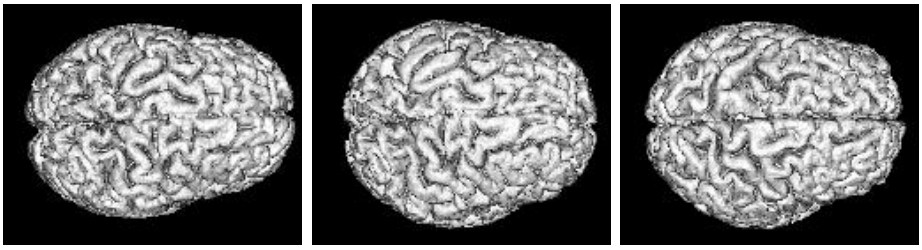

Fig. 18. An experiment with two real brain data. From left to right: 1) the reference brain; 2) warped reference brain (warped to match the target); 3) the target brain. Note how the warping changes the brain's global shape.

4.2 Experiments on Real Data

We conducted some preliminary experiments on a few pairs of anatomical MRI brain data. At the present stage, our feature representation is well suited for the global alignment of the brain. We include one example based on the currently available feature representation here in Figure 18. We believe that to provide a more adequate representation of the complex brain structures, more detailed brain structural features are needed, especially from the sub-cortical regions.

5 Discussion and Conclusion

The above experiments show that the combination of different features inherits each feature's merits while avoiding the problems stemming from exclusive usage. As we discussed above, though the outer cortex provides a good model for the overall brain shape, its ellipsoidal shape is vulnerable to rotation errors. This weakness can be eliminated with the help of the sulcal ribbons. On the other hand, without the outer cortical surface, the sulcal ribbons alone are too sparse. By jointly registering the features, the alignment is better constrained. The new joint clustering and matching (JCM) method provides a simple and effective way of accomplishing the goal of unified registration.

Even though we only discussed using multiple surfaces in this paper, the idea of fusing different features into a common point representation space is general and can be easily applied to other features such lines and curves. It is also possible to add weight factors and attributes to each type of feature in this framework. The point clustering-matching engine is quite general as well. The matching of the feature points is indirectly accomplished by matching cluster centers. In essence, we are trying to achieve a many-to-many fuzzy matching between the original dense feature point-sets. For the discrete point representation derived through sub-sampling, exactly corresponding points are rare.

Therefore, perfect one-to-one matching may no longer be the optimal solution. The fuzziness in our matching greatly alleviates this problem by allowing partial correspondences. A single parameter T afford fine control over the degree of such fuzziness. If we stop the algorithm at a higher T, higher fuzziness is achieved. If we lower the final temperature, we can achieve the limit of binary correspondence which may or may not be desirable. Clearly, the effect of varying the two free parameters—the number of cluster centers and the regularization parameter—need to be studied. Also, we speculate that the inclusion of additional information such as the principal components of the covariance matrix of each cluster may improve the performance of the JCM algorithm especially on larger data-sets.

References

Bajcsy, R., Kovacic, S., 1989. Multiresolution elastic matching. Computer Vision, Graphics and Image Processing 46, 1–21.

Bookstein, F. L., June 1989. Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Trans. Patt. Anal. Mach. Intell. 11 (6), 567–585.

Camion, V., Younes, L., 2001. Geodesic interpolating splines. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2001). Springer, pp. 513–527.

Christensen, G., 1999. Consistent linear-elastic transformations for image matching. In: Proceedings of Information Processing in Medical Imaging—IPMI 1999. Springer-Verlag, New York, pp. 224–237.

Christensen, G., Joshi, S., Miller, M., 1997. Volumetric transformation of brain anatomy. IEEE Trans. Med. Imag. 16 (6), 864–877.

Chui, H., Rambo, J., Duncan, J., Schultz, R., Rangarajan, A., 1999. Registration of cortical anatomical structures via robust 3D point matching. In: Proceedings of Information Processing in Medical Imaging–IPMI 1999. Springer, New York, pp. 168–181.

Chui, H., Rangarajan, A., 2000. A new algorithm for non-rigid point matching. In: Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Vol. 2. IEEE Press, pp. 44–51.

Collins, D., Goualher, G., Evans, A., 1998. Non-linear cerebral registration with sulcal constraints. In: Wells, W., Colchester, A., Delp, S. (Eds.), Medical Image Computing and Computer-Assisted Intervention. Vol. 1496 of Lecture Notes in Computer Science. Springer, pp. 974–984.

Collins, D., Holmes, C., Peters, T., Evans, A., 1995. Automatic 3D model-based neuro-anatomical segmentation. Human Brain Mapping 3 (3), 190–208.

Davatzikos, C., 1997. Spatial transformation and registration of brain images using elastically deformable models. Computer Vision and Image Under-

standing: Special Issue on Medical Imaging 6 (2), 207–222.
Davatzikos, C., Prince, J., 1994. Brain image registration based on curve mapping. Proc. of the IEEE Workshop on Biom. Image Anal. , 245–254.
Gee, J., 1995. Probabilistic matching of deformed images. Ph.D. thesis, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, PA.
Gold, S., Rangarajan, A., Lu, C. P., Pappu, S., Mjolsness, E., 1998. New algorithms for 2-D and 3-D point matching: pose estimation and correspondence. Pattern Recognition 31 (8), 1019–1031.
Hathaway, R., 1986. Another interpretation of the EM algorithm for mixture distributions. Statistics and Probability Letters 4, 53–56.
Hofmann, T., Buhmann, J. M., Jan. 1997. Pairwise data clustering by deterministic annealing. IEEE Trans. Patt. Anal. Mach. Intell. 19 (1), 1–14.
Joshi, S., Miller, M., 2000. Landmark matching via large deformation diffeomorphisms. IEEE Trans. Image Processing 9, 1357–1370.
Khaneja, N., Miller, M. I., Grenander, U., 1998. Dynamic programming generation of curves on brain surfaces. IEEE Transaction on Pattern Analysis and Machine Intelligence 20 (1), 1260–1265.
MacDonald, D., Kabani, N., Avis, D., Evans, A., 2000. Automated 3d extraction of inner and outer surfaces of cerebral cortex from mri. NeuroImage 12, 340–356.
McLachlan, G. J., Basford, K. E., 1988. Mixture models: inference and applications to clustering. Marcel Dekker, New York.
Meyer, C., Boes, J. L., Kim, B., Bland, P. H., 1997. Demonstration of accuracy and clinical verstility of mutual information for automatic multimodality image fusion using affine and thin plate spline warped geometric deformations. Medical Image Analysis 1 (3), 195–206.
Rangarajan, A., Chui, H., Mjolsness, E., Pappu, S., Davachi, L., Goldman-Rakic, P., Duncan, J., 1997. A robust point matching algorithm for autoradiograph alignment. Medical Image Analysis 4(1), 379–398.
Redner, R. A., Walker, H. F., 1984. Mixture densities, maximum likelihood and the EM algorithm. SIAM Review 26 (2), 195–239.
Rose, K., Gurewitz, E., Fox, G., Aug. 1990. Statistical mechanics and phase transitions in clustering. Physical Review Letters 65 (8), 945–948.
Sandor, S., Leahy, R., 1997. Surface based labeling of cortical anatomy using a deformable atlas. IEEE Trans. Med. Imag. 16 (1), 41–54.
Thompson, P., MacDonald, D., Mega, M., Holmes, C., Evans, C., Toga, A., 1997. Detection and mapping of abnormal brain structure with a probabilistic atlas of cortical surfaces. Journal of Computer Assisted Tomography 21 (4), 567–581.
Thompson, P., Toga, A. W., August 1996. A surface-based technique for warping three-dimensional images of the brain. IEEE Trans. Med. Imag. 5 (4), 402–417.
Toga, A., Mazziotta, J., 1996. Brain Mapping: The Methods. Academic Press.
Vaillant, M., Davatzikos, C., 1999. Hierarchical matching of cortical features

for deformable brain image registration. In: Proceedings of Information Processing in Medical Imaging–IPMI 1999. Vol. 1613. Springer-Verlag, New York, pp. 182–195.

Vaillant, M., Davatzikos, C., Bryan, R., 1996. Finding 3D parametric representations of the deep cortical folds. In: Amini, A., Bookstein, F. L., Wilson, D. (Eds.), Proc. of the Workshop on Mathematical Methods in Biomedical Image Analysis. IEEE Computer Society Press, pp. 151–159.

Wahba, G., 1990. Spline models for observational data. SIAM, Philadelphia, PA.

Wang, Y., Staib, L. H., July 2000. Boundary finding with prior shape and smoothness models. IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (7), 738–743.

Xu, C., Pham, D., Prince, J. L., 1998. Reconstruction of the central layer of the human cerebral cortex from mr images. In: Proceedings of Medical Image Computing and Computer-Assisteed Intervention–MICCAI 98. pp. 481–488.

Yuille, A. L., Grzywacz, N. M., June 1989. A mathematical analysis of the motion coherence theory. Intl. J. Computer Vision 3 (2), 155–175.

Yuille, A. L., Stolorz, P., Utans, J., March 1994. Statistical physics, mixtures of distributions, and the EM algorithm. Neural Computation 6 (2), 334–340.

Zeng, X., Staib, L. H., Schultz, R. T., Duncan, J. S., 1999a. Segmentation and measurement of the cortex from 3D MR images using coupled-surfaces propagation. IEEE Transaction on Medical Imaging 18 (10), 927–937.

Zeng, X., Staib, L. H., Schultz, R. T., Tagare, H., 1999b. A new approach to 3D sulcal ribbon finding from MR images. In: MICCAI 99. pp. 148–157.