Fast Brain Matching with Spectral Correspondence

Herve Lombaert[1,2], Leo Grady[2], Jonathan R. Polimeni[3], Farida Cheriet[1]

[1] Ecole Polytechnique de Montreal, Canada
[2] Siemens Corporate Research, Princeton, NJ
[3] Athinoula A. Martinos Center for Biomedical Imaging, Department of Radiology, Harvard Medical School, Massachusetts General Hospital, Charlestown, MA

Abstract. Brain matching is an important problem in neuroimaging studies. Current surface-based methods for cortex matching and atlasing, although quite accurate, can require long computation times. Here we propose an approach based on spectral correspondence, where spectra of graphs derived from the surface model meshes are matched. Cerebral cortex matching problems can thus benefit from the tremendous speed advantage of spectral methods, which are able to calculate a cortex matching in seconds rather than hours. Moreover, spectral methods are extended in order to use additional information that can improve matching. Additional information, such as sulcal depth, surface curvature, and cortical thickness can be represented in a flexible way into graph node weights (rather than only into graph edge weights) and as extra embedded coordinates. In control experiments, cortex matching becomes almost perfect when using additional information. With real data from 12 subjects, the results of 288 correspondence maps are 88% equivalent to (and strongly correlated with) the correspondences computed with FreeSurfer, a leading computational tool used for cerebral cortex matching. Our fast and flexible spectral correspondence method could open new possibilities for brain studies that involve different types of information and that were previously limited by the computational burden.

1 Introduction

The human cerebral cortex is composed of many distinct brain areas whose locations relative to the folding pattern are highly stereotyped. In many neuroimaging studies, finding corresponding locations between two individuals allows data to be pooled across subjects and enables the investigation of functional and anatomical differences between individuals. Early attempts at computing correspondences relied on the extrinsic geometry seen on brain volumetric images. In 1967, Talairach *et al.* [19] introduced an early version of a brain atlas in the form of a 3D stereotaxic coordinate system. Despite its popularity, this method matched volumetric brain data using 3D Euclidean distances, which ignored geometric variabilities in the folding pattern. Techniques based on high-dimensional deformations allow for the alignment of volumetric brain image data. However, the lack of an explicit model for the brain surface often creates misaligned cortical areas [1]. Later, it was demonstrated that *surface-based alignment* [6,5,4,22,20], which operates by directly registering surface models of the cerebral cortex, significantly outperforms volume-based approaches [6,5]. The success of these surface-based techniques depends on the stability of the

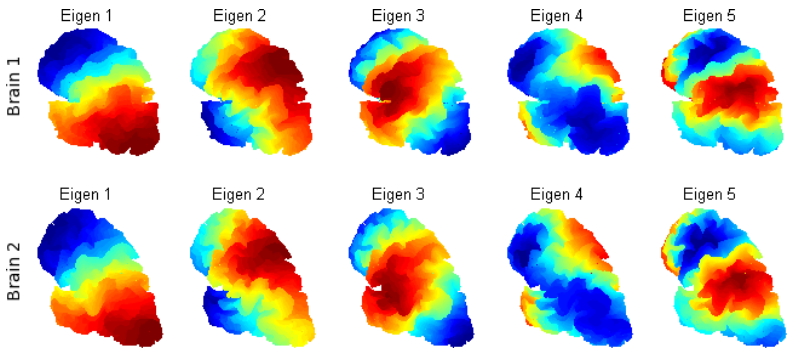


Fig. 1. Eigendecomposition of the graph Laplacian (using (2)) showing the first six eigenvectors (or six spectral components, one per column) of two brain surfaces (top and bottom rows). The color indicates a spectral coordinate for each point.

folding pattern across subjects. Some cortical areas are indeed consistently in a fixed position relative to the folding pattern [6,11], whereas other areas float around and do not seem well correlated to the folding pattern. These areas, however, may be correlated with measurable anatomical features other than the explicit cortical geometry. One successful method for computing brain surface correspondences was introduced by Fischl *et al.* [6]. It inflates each cerebral hemisphere surface to a sphere by a process that minimizes metric distortions, thus preserving local distances and areas. The cortical surface model is then mapped to the sphere using a nonrigid deformation driven by geometric features of the original folding pattern. This is the method used by FreeSurfer, a leading and widely used tool for brain surface reconstruction, matching, and atlasing. Despite its accuracy, FreeSurfer suffers from a severe computational burden, which causes it to be very slow—it can take hours to compute a correspondence map between two cerebral cortices.

In order to alleviate this severe computational burden, we introduce a different approach for brain surface matching based on spectral correspondence. Spectral methods [3] present the tremendous advantage of being extremely fast—on the order of seconds. Correspondences are found on a graph *spectrum*, which is essentially the eigendecomposition of the graph Laplacian of an underlying shape model (illustrated on Fig. 1). Spectral methods have already been used in many fields, including in computer vision with the segmentation and registration of shapes in images [15], and recently in medical applications with the analysis of brain shape features [14,16] and with the smoothing of cortical surfaces [2]. Umeyama [21] and later Scott and Longuet-Higgins [17], pioneered the use of spectral methods for the correspondence problem. Shapiro and Brady [18] compared ordered eigenvectors of a proximity matrix to find correspondences. Their work served as a basis for future spectral correspondence methods. Variants includes the use of different proximity matrices using different kernels, the use of the adjacency matrix, or different normalized Laplacian matrices. Mateus *et al.* [12] proposed an original unsupervised spectral method with an alternative to eigenvalue ordering based on eigenvectors histograms and refining the eigenvectors alignment with a probabilistic point matching. Jain and Zhang [10] tackle the eigenvectors alignment with a nonrigid deformation based on thin plate splines.

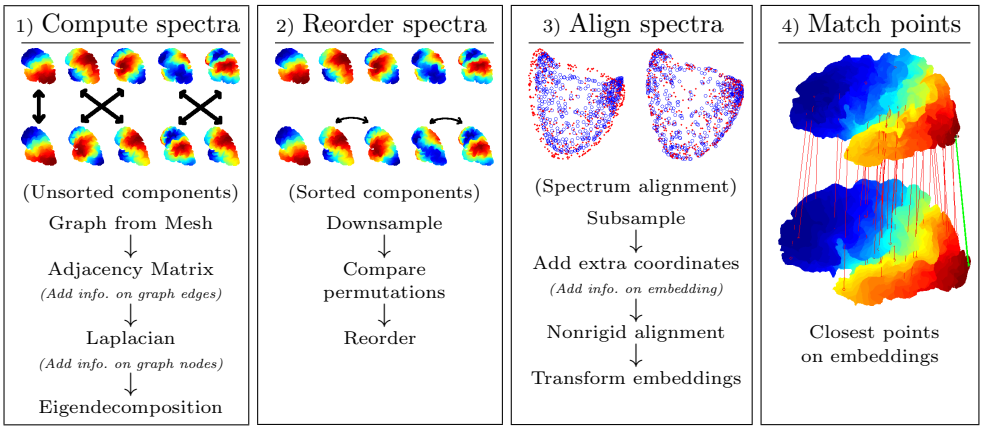


Fig. 2. Algorithm summary: *First*, we build two graphs and set the graph edges (the adjacency matrix) and on the graph nodes (the Laplacian matrix). The eigendecomposition of the graph's Laplacian reveals the spectral components. *Second*, we reorder the components by finding the optimal permutation of components. *Third*, we deform the spectral embeddings. *Finally*, matching points are found with closest points in both spectral representations.

Previous spectral correspondence methods employ solely geometric information by weighting the graph edges with the distances between connected pairs of vertices. However, in order to use certain quantities (like sulcal depth, surface curvature, or cortical thickness), we must modify the spectral correspondence approach to incorporate information beyond edge length. To our knowledge, we are the first to present the use of *node weighting* in a spectral correspondence method. Additional information can indeed be incorporated into the Laplace operator, which implicitly contains metric information about nodes and edges. Moreover, additional information can be used as extra embedded coordinates when aligning the eigenvectors. This added level of flexibility makes our method a good candidate for brain studies involving various types of information with a large number of subjects.

After detailing our approach in the next section, we show in a control experiment that additional information can dramatically improve the performance of a spectral method. Using data from 12 subjects, we validate our method by comparing the computed correspondences with those generated by FreeSurfer [6]. We show that our method produces results, in a fraction of time required by FreeSurfer, that approach the accuracy of FreeSurfer. We believe that this large gain in processing speed would open the doors to new brain studies that were previously limited by the computational burden of the cortex matching calculation. Therefore, this method has the potential to be a significant tool for use in neuroscience.

2 Method

The proposed algorithm finds correspondences by comparing cortex representations, called *spectra* (illustrated on Fig. 1). The spectrum of a brain surface mesh is independent of its extrinsic geometry. In order to adapt a spectral method to brain surface matching, we must solve for several issues. First, we show how additional information (sulcal depth [6], surface curvature, and cortical thickness) can be used

in spectral methods as weightings of the *graph nodes*. Second, we improve the ordering of the spectral components by finding an optimal permutation of the underlying eigenvectors. And third, we align the spectra in a multidimensional space using a nonrigid point transformation method. Our algorithm is summarized in Figure 2.

2.1 Spectral correspondence

Given a shape defined by its collection of nodes $\mathcal{X} = \{x_1, x_2, ..., x_n\}$ and a neighborhood system $\mathcal{N}$ connecting a subset of these nodes (e.g., a mesh with vertices and faces), it is possible to construct its corresponding graph $\mathcal{G}_\mathcal{X}$. Here, we define the adjacency matrix W in terms of *affinity* weights (see [8]), which are derived from a distance metric $\text{dist}(x_i, x_j)$ between two neighboring vertices (x_i, x_j) (ε is a small penalizing term):

$$W_{ij} = \begin{cases} (\text{dist}(i,j) + \varepsilon)^{-1}, & \text{if } i \in \mathcal{N}(j), \text{and } i \neq j \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

It was shown in [8] that the general Laplacian operator on a graph takes the form $\tilde{L} = GL = G(D - W)$, where D is a diagonal matrix defined as $D_{ii} = \sum_j W_{ij}$ and G is the diagonal matrix of node weights. Typically in spectral correspondence, G is set to $G = D^{-1}$. However, we propose here to replace the default assignment $G = D^{-1}$ with *any* meaningful node weighting. Therefore, data associated with a mesh, such as distances or other additional information, can be incorporated in a graph on either its edges (in (1)), or its nodes by manipulating G.

We assume that the sulcal depth at each point, $\{s_1, s_2, ..., s_n\}$, the Gaussian curvature at each point, $\{\kappa_1, \kappa_2, ..., \kappa_n\}$, or the cortical thickness at each point, $\{t_1, t_2, ..., t_n\}$, are pointwise characteristics and should be primarily defined on the graph nodes rather than on edges. We use the exponential of the positive diagonal matrices $\exp(S) = \exp(\text{diag}(s_1, s_2, ..., s_n))$, and $\exp(K) = \exp(\text{diag}(\kappa_1, \kappa_2, ..., \kappa_n))$, and $\exp(T) = \exp(\text{diag}(t_1, t_2, ..., t_n))$. We propose to incorporate additional information in the weighting of the nodes by defining the graph Laplacian as:

$$\tilde{L} = GL, \text{ where } G = D^{-1}\left(c_s \exp(S) + c_\kappa \exp(K) + c_t \exp(T)\right)^{-1}, \tag{2}$$

where c_s, c_κ, and c_t are weighting factors. We use for instance $c_s = \text{mean}\{D_{ii}\}_{i=1\cdots n} / \text{mean}\{\exp(s_i)\}_{i=1\cdots n}$. The right eigenvectors of the Laplacian comprise the graph spectrum $\hat{X} = \{\hat{X}^{(1)}, \hat{X}^{(2)}, ..., \hat{X}^{(n)}\}$. Figure 1 shows an example of spectral components for two brain hemispheres where each column depicts a different spectral component. Each eigenvector $\hat{X}^{(u)}$ represents a different (weighted) harmonic on a mesh surface that represents an intrinsic geometric property. The values $\hat{x}_i^{(u)}$, $i \in [1 \ldots n]$, give the *spectral coordinates* for each point x_i. Eigenvectors associated with the lower non-zero eigenvalues (e.g., $\hat{X}^{(2)}, \hat{X}^{(3)}$) represent coarse (low-frequency) intrinsic geometric properties of the shape, the first of them $\hat{X}^{(2)}$ is called the *Fiedler vector*, while eigenvectors associated with higher eigenvalues (e.g., $\hat{X}^{(n-1)}, \hat{X}^{(n)}$) represent fine (high-frequency) geometric properties. The core idea of our method is to match two meshes $\mathcal{X}$ and $\mathcal{Y}$ by comparing their corresponding spectra $\hat{X}$ and $\hat{Y}$ rather than directly comparing the meshes themselves.

2.2 Ordering the spectra

Each point of the brain surface mesh is represented with K spectral components associated with the K smallest eigenvalues, i.e., the embedded representations are $\hat{X}^K = [\hat{X}^{(2)}, ..., \hat{X}^{(K+1)}]$ and $\hat{Y}^K = [\hat{Y}^{(2)}, ..., \hat{Y}^{(K+1)}]$. Unfortunately, the spectral coordinates of the two meshes may not be directly comparable as a result of two phenomena. First, there exists a *sign ambiguity* when computing eigenvectors (i.e., if $Ax = \lambda x$ then $A(-x) = \lambda(-x)$), requiring a sign check of each eigenvector in the two meshes. Additionally, as a result of greater algebraic multiplicity of an eigenvalue, it may be possible that the *ordering* of the lowest eigenvectors will change, e.g., if two eigenvectors correspond to the same eigenvalue in both meshes, then the solver may compute these eigenvectors in one order for the first mesh and in the opposite order for the second mesh. For large meshes, this is a recurrent problem and the eigenvectors must be reordered. Since different brains do not present major discrepancies or major articulated deformations between individuals, the eigenvectors may be reordered by comparing their values at all pairs of closest points between the two brain hemispheres.

To speed up the reordering, all eigenvectors are first subsampled by selecting randomly a few points (we use 500 points in our experiments). Their spectral coordinates are normalized between 0 and 1 and denoted as $\hat{x}^{(i)}$. A spatial integration of all differences within pairs of closest points provides a similarity measure, i.e., if the eigenvectors $\hat{x}^{(i)}$ and $\hat{y}^{(j)}$ correspond to each other in both meshes, for all closest Cartesian points $\{(x_i, y_{i'})\}_{i=1\cdots n}$, the difference of their associated spectral coordinates are computed. All the differences of potentially corresponding eigenvectors, $\hat{x}^{(u)}$ and $\hat{y}^{(v)}$, are gathered in a dissimilarity matrix, $C(\hat{x}^{(u)}, \hat{y}^{(v)}) = \sum_{i=1}^{N} \left(\hat{x}_i^{(u)} - \hat{y}_{i'}^{(v)}\right)^2$, where $y_{i'} \in \mathcal{Y}$ is closest to $x_i \in \mathcal{X}$. The Hungarian algorithm may be used to find an optimal permutation of eigenvectors $\hat{y}^{(v)}$ and, in order to remove the sign ambiguity, the minimal dissimilarity between the comparison of $\hat{x}^{(u)}$ and $\hat{y}^{(v)}$, and $\hat{x}^{(u)}$ and $-\hat{y}^{(v)}$ is used. The cost matrix used in the Hungarian algorithm is $Q(u, v) = \min\{C(\hat{x}^{(u)}, \hat{y}^{(v)}), C(\hat{x}^{(u)}, -\hat{y}^{(v)})\}$. After permutation, any eigenvector $\hat{x}^{(u)}$ corresponds with $\hat{y}^{(u)}$ and has a permutation cost $C^{(u)}$.

2.3 Alignment of Spectra

After reordering and handling the sign ambiguity, the eigenvectors of the two meshes may be assumed to have the same ordering in both embeddings (i.e., $\hat{x}^{(u)}$ corresponds with $\hat{y}^{(v)}$). However, the embedded representations, $\hat{X}^K$ and $\hat{Y}^K$, need to be aligned (as illustrated in the third box of Fig. 2, both spectra have slight differences) so that closest points in these embedded representations would reveal corresponding points in both shapes (i.e., if $\hat{y}_j^K$ is the closest point to $\hat{x}_i^K$, then x_i corresponds with y_j).

In order to perform this alignment, each eigenvector $\hat{x}^{(u)}$ is first weighted with $\exp(-(C^{(u)}\lambda_{\hat{x}^{(u)}})^2/2\sigma^2)$, where $C^{(u)}$ is the permutation cost, $\lambda_{\hat{x}^{(u)}}$ is its associated eigenvalue, and σ is a penalizing factor, we use $\sigma = \text{mean}\left\{C^{(u)}\lambda_{\hat{x}^{(u)}}\right\}_{u=1\cdots K}$. Low-frequency eigenvectors, associated with coarser geometric properties (i.e., small eigenvalues $\lambda_{\hat{x}^{(u)}}$), will thus have more importance than the high-frequency eigenvectors associated with finer details, and pairs of corresponding eigenvectors will have more importance if they have strong similarities (i.e., low permutation costs $C^{(u)}$).

To further constrain the alignment of the mesh representations, we add extra information as additional coordinates to the embedded representation. Specifically, we concatenate with our first K spectral components $\hat{X}^K = [\hat{X}^{(2)}, ..., \hat{X}^{(K+1)}]$ the extra coordinates, $\exp(\mathbf{S})$, $\exp(\mathbf{K})$, and $\exp(\mathbf{T})$. These extra components are also normalized to be comparable with the spectral coordinates. The embedded representations to be aligned are thus:

$$\begin{aligned}\tilde{X} &= [\hat{X}^K, c_s \exp(\mathbf{S}_{\mathcal{X}}), c_\kappa \exp(\mathbf{K}_{\mathcal{X}}), c_t \exp(\mathbf{T}_{\mathcal{X}})], \text{ and,} \\ \tilde{Y} &= [\hat{Y}^K, c_s \exp(\mathbf{S}_{\mathcal{Y}}), c_\kappa \exp(\mathbf{K}_{\mathcal{Y}}), c_t \exp(\mathbf{T}_{\mathcal{Y}})].\end{aligned} \tag{3}$$

The alignment of the spectral components can be viewed as a nonrigid registration, $\tilde{X}^K = \Phi(\tilde{Y}^K)$. The third box of Fig. 2 shows the alignment challenge where the first three spectral components $(\hat{X}^{(2)}, \hat{X}^{(3)}, \hat{X}^{(4)})$ are used as 3D (x, y, z) coordinates for visualization purposes. The Robust Point Matching with a thin plate spline-based transformation is often used for 2D or 3D registration. However, with this approach, the final registration depends on the number and choice of the control points. We apply the Coherent Point Drift method [13] which is fast and demonstrates excellent performance in this application. To increase speed in our algorithm, we subsample $\tilde{X}$ and $\tilde{Y}$ by taking randomly a few points (we used 500 points). The Coherent Point Drift method finds a continuous transformation Φ that can be applied on all points of $\tilde{Y}$. After aligning both embedded representations (i.e., $\tilde{X} = \Phi(\tilde{Y})$), it is possible to directly compare them, i.e. two points which are closest in the embedded representations, $\tilde{x}_i$ and $\tilde{y}_{i'}$, are treated as corresponding points in the meshes $\mathcal{X}$ and $\mathcal{Y}$. The fourth box of Fig. 2 illustrates a few pairs of corresponding points. All pairs of points connected by the red lines have the closest embedded coordinates.

3 Results

Our methodology introduces several new concepts for spectral methods and shows how these methods may be customized for the application of cerebral cortex matching. We first show in an intuitive experiment the effect of node weighting on a simple mesh. Second, we measure the accuracy of spectral methods on a controlled experiment with a known ground truth. Third, we analyze the accuracy of our method against FreeSurfer. For this comparison we used 24 cerebral hemispheres from 12 subjects, and based our comparison on 288 matches using different combinations of additional information. Each brain surface mesh has been extracted using FreeSurfer from T_1-weighted magnetic resonance images.

3.1 Node Weighting

We believe that we are the first to utilize node weights in a spectral correspondence approach. Consequently, we briefly devote some space to give an intuition about the behavior of these node weights in the context of spectral correspondence. To demonstrate the differences in weighting the edges and the nodes, we choose to show the Fiedler vector on a spherical mesh with an asymmetric vertex distribution. The concentration of vertices at the poles guides the spectral eigendecomposition (i.e., the Fielder vector is aligned with the sphere poles as shown in top-left sphere of Fig.

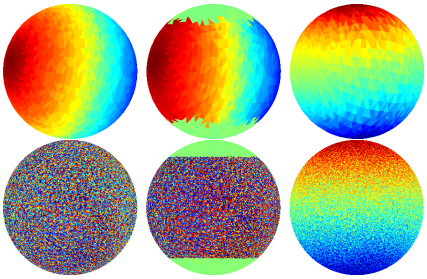


Fig. 3. Effect of node weighting. *Top row*, three cases showing the Fiedler vector on spheres of same orientation. *Left*, with no weighting: the vector is aligned with the sphere axis. *Middle*, with *edge* weighting: the top and bottom sections of the sphere are masked. *Right*, with *node* weighting: the vector is reorientated vertically. *Bottom row*: Same three experiments ran on 100 randomly orientated spheres. Their Fiedler vectors are accumulated. The reorientation of the Fiedler vector via node weighting (*bottom-left*) is clearer.

3). The accumulation of Fiedler vector on randomly orientated spheres should yield a uniform distribution (as shown in the bottom-left sphere of Fig. 3). We show that encoding weights on nodes can influence the orientation of the Fiedler vector. For instance, to reorient the Fiedler vector vertically, we use a texture data where the top section and the bottom sections of a sphere (on a world z axis) are heavily weighted. For all points $\{x_i\}_{i=1\cdots n}$, their weights are $\Theta = [\theta_1, \theta_2, ..., \theta_n]$ where $\theta_i = 1000$ if $|x_i^{(z)}| > 0.4$ (i.e., a large weighting at the top and bottom of the sphere along the z axis) and $\theta_i = 1$ if $|x_i^{(z)}| \leq 0.4$ (i.e., a low weighting around the middle section).

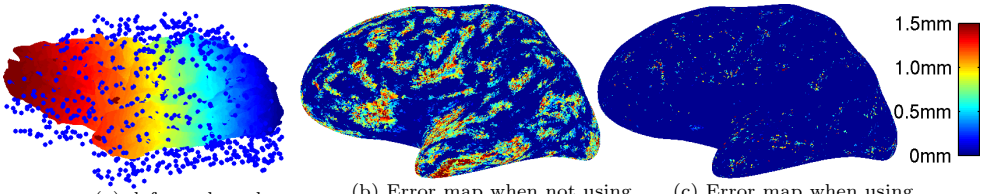
In previous approaches to spectral correspondence, weights are encoded on graph edges. The texture weight Θ is added to the edge weights (1): $w_{ij} = (\text{dist}(x_i, x_j) + \varepsilon)^{-1} \times (|\theta_i - \theta_j| + \varepsilon)^{-1}$. The resulting weighting will highlight the texture boundaries isolating three distinct regions: the top, the middle, and the bottom section. This creates a multiplicity of three in the zero eigenvalues, and as shown in the top-middle sphere of Fig. 3, the Fiedler vector covers the largest section.

In our method, we propose to weight nodes in a graph in addition to weighting the edges. In order to compare with the previous experiment, we do not incorporate the texture Θ on graph edges. It is used on graph nodes (2): $G = D^{-1}\text{diag}\,(\Theta)^{-1}$. After the spectral decomposition, the multiplicity of the zero eigenvalue is 1 (i.e., there is one distinct object), and the Fiedler vector is aligned with the texture (i.e., with the world z axis). To verify this alignment, we repeated this experiment with 100 spheres orientated randomly and we accumulated the values of the Fiedler vectors (bottom row of Fig. 3). The principal axis of the accumulated values, $(-0.04, 0.12, 0.99)$, is indeed vertical when weighting the graph nodes (bottom-right of Fig. 3). The accumulation of the Fiedler vectors does not show an apparent principal axis when weighting only the graph edges (bottom-middle of Fig. 3), or when the texture data is not used (bottom-left of Fig. 3).

These three experiments show that weighting the graph nodes provides a new way for incorporating additional pointwise information and behaves differently than weighting the graph edges. Spectral methods can thus benefit from this idea, not only in cortex matching, but in various applications. The next experiment shows how node weighting improves matching in spectral methods.

3.2 Matching Deformed Brain Surfaces

Every individual has a unique folding pattern in the cerebral cortex, however there are many large-scale similarities. Before considering inter-subject cortex matching,



(a) deformed meshes (b) Error map when not using additional information (c) Error map when using additional information

Fig. 4. Comparison with ground truth. *Left*: The deformed mesh (colored with its Fielder vector) overlaid with the original vertex positions illustrated by the blue dots. *Middle*: The mean distance error is 0.44 mm with a standard spectral correspondence method (i.e., without additional information). *Right*: With additional information (sulcal depth, surface curvature, and cortical thickness), the error drops to 0.05 mm.

we study the intra-subject case. We do so with a controlled, simulated deformation to analyze how additional information improves the accuracy of our method, and how the additional information should be incorporated. For our experiment, we match one brain hemisphere with a deformed version of itself. The vertex indexing remains the same in the deformed version, therefore the true matching is known (i.e., for all i, vertex p_i in the first mesh should match the vertex q_i in the second mesh). We severely deform one of the cortical hemisphere surface models with the transformation $q^{(z)} = (1+\alpha)p^{(z)}$ (a compression in the z-axis controlled by α) and $q^{(x)} = p^{(x)} + \beta r^2 / \max(r^2)$ with $r^2 = {p^{(x)}}^2 + {p^{(y)}}^2$ (a radial distortion controlled by β). This simulates a deformation due to a drastic change in the head shape. The deformation however preserves the same mesh topology (i.e., with no discontinuity and no intersecting faces). Fig. 4 illustrates the position of the original hemisphere with the blue dots and the deformed hemisphere with the colored mesh. We quantify the accuracy of our matching by measuring the mean distance between all points and their corresponding matched points. That is, for all points p_i in mesh 1 matching q_j in mesh 2, we average the distance: $\text{mean}\,(\text{dist}(p_i, p_j))$. When no additional information is used, as it is the case in most state-of-the-art spectral methods, we find an average error distance of 0.44 mm as shown in the first error map of Fig. 4. Most errors are actually located on the sulci extrema.

Additional information can be incorporated as *node* weighting by using (2); as *edge* weighting by similarly adding additional term to (1) such as $w_{ij} = 1/(d(i,j) + \epsilon)\exp(-(s_i - s_j)^2/2\sigma_s^2)\exp(-(\kappa_i - \kappa_j)^2/2\sigma_\kappa^2)\exp(-(t_i - t_j)^2/2\sigma_t^2)$, where $\sigma_{s,\kappa,t}$ are penalizing factors; or as additional coordinates in the alignment process by using (3). Three sources of additional information (sulcal depth, surface curvature, and cortical thickness) can thus be used in three different ways in our method. That is 512 possible combinations ($2^{3\times3}$). We iterate through all of them and found that adding information as additional coordinate has the strongest impact on the accuracy of our matching. Adding the sulcal depth as the only additional feature yields an error of 0.16 mm; adding only the surface curvature yields an error of 0.35 mm; and the cortical thickness yields an error of 0.14 mm. Adding single or multiple source of additional information on only the graph nodes does not yield significant improvement (0.44 mm), nor does representing this additional information on only the graph edges (0.44 mm). However, by adding all three features and using all of them on the graph nodes and on the graph edges, the error drops to 0.06 mm. Our

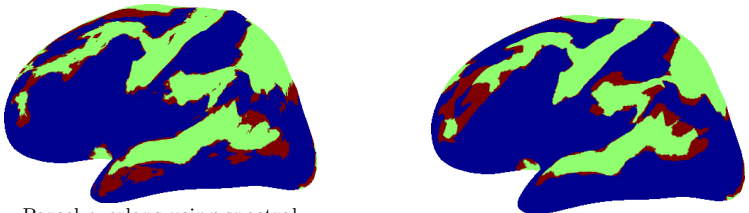


Fig. 5. In *green*, the areas where the projected sulcal regions of two cortices overlap, and in *red* the projection mismatches. *(Left brain)* Correspondences were computed in 124 seconds using our method, while *(right brain)* FreeSurfer required several hours.

best combination of additional information was obtained when using all three features and when using sulcal depth and cortical thickness on the graph nodes, yielding an error of 0.05 mm. The error map in the right of Fig. 4 shows an almost perfect matching with our best-performing combination.

3.3 Validation

Brain surface matching is an ambiguous problem. Indeed, sulci morphology and topology differ from one individual to another. There is no ground truth available for perfect brain surface matching. However, FreeSurfer [6] has been demonstrated to provide highly accurate cortical matchings that closely align cortical areas across subjects [9] and therefore provides a reliable benchmark for our comparison. The delineations of 81 sulcal regions are available for 24 hemispheres (12 subjects). These sulcal regions were obtained using an automatic parcellation of the cortex [7] and are considered as our gold standard. Although folding parcellations are not excepted to align between subjects in all cases (except for the primary folds), they do provide means to compare the two methods. We use correspondence maps generated by FreeSurfer and by our method to project the parcellation areas onto different brain hemispheres and we measure their overlaps (illustrated on Figure 5). To process a mesh of 135,000 vertices, FreeSurfer has a varying processing time which is always on the order of several hours, while the time required by our method is just on the order of a few minutes. To process all our 288 possible pairs of left and right brain hemispheres, our method required on average 124 seconds on a 2.8 GHz Intel Pentium 4 using unoptimized Matlab code (with meshes of 20,000 vertices, our method performed in 19 seconds). The code could benefit further from parallel programming and the use of GPU. The total time was 9 hours on a single computer, a substantial advantage compared to the several weeks required by FreeSurfer to process all 288 cortex matchings in series. Each overlap ratio is defined by the ratio of set intersection to set union. Figure 6 shows the overlap ratios for the largest sulcal parcellations using our method (our best setting is shown in cyan) and FreeSurfer (red). Our results are consistent across all 81 sulcal regions (i.e., whenever FreeSurfer outputs a higher overlap ratio, our method also consistently outputs a higher overlap ratio). Specifically, our results are correlated to FreeSurfer's overlaps with a correlation coefficient of $\rho = 0.816$. When comparing larger regions[1] (illustrated on Fig. 6: parcels 9, 10, 18, 23, 24, 26, 41, 42,

[1] Sulcal regions: 9 (*G frontal middle*), 10 (*G frontal middle*), 18 (*G occipit temp med Lingual part*), 23 (*G parietal inferior Supramarginal part*), 24 (*G parietal superior*), 26

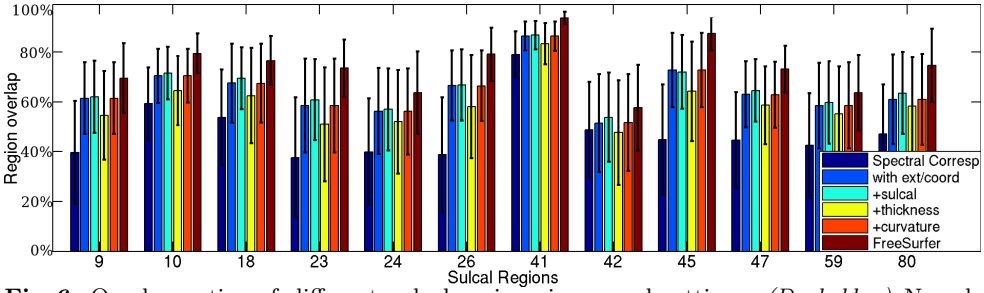


Fig. 6. Overlap ratios of different sulcal regions in several settings. *(Dark blue)* No additional information (65% of FreeSurfer's performance), *(blue)* all additional information (sulcal depth, cortical thickness and cortical curvature) as extra coordinates (87%), *(cyan)* all additional information as extra coordinates and *sulcal depth* on graph nodes and graph edges (88%), *(yellow)* all additional information as extra coordinates and *cortical thickness* on graph nodes and graph edges (78%), *(orange)* all additional information as extra coordinates and *cortical curvature* on graph nodes and graph edges (87%) and *(red)* by FreeSurfer (requiring weeks of computations). Our method only requires 9 hours and is strongly correlated with FreeSurfer (correlation coefficient of $\rho = 0.816$). The error bars show the standard deviation of each overlap ratio.

45, 47, 59, and 80, as defined in [7]), FreeSurfer's overlap ratios are on average 74%. In its best setting (using sulcal depth as additional information), our method gives 88% that of FreeSurfer's overlap ratios.

3.4 Combination of Additional Information

Besides information on sulcal depth, we had access to information on cortical thickness and on surface curvature. The cortical thickness is another clinically relevant anatomical measure, which is calculated by FreeSurfer from anatomical MRI images. The sulcal curvature is estimated with the Gaussian curvature ($\kappa_1\kappa_2$) of the mesh. We first analyze the performance of our method using five configurations of different combinations of additional features. For each configuration, we ran our method on the 288 pairs of brain hemispheres (totaling 1440 matchings). The results are summarized in Figure 6. The *first* configuration uses no additional information ($G = D^{-1}$ in (2)). In that configuration, the average overlap ratio on the largest parcels is only 48% (in comparison, FreeSurfer performs at 74%). In the *second* configuration, we use sulcal depth, cortical thickness, and cortical curvature as extra coordinates in the spectral alignment (using (3)). The average overlap ratio increases to 64% (or 87% of FreeSurfer's performance), a 34% increase from the previous configuration. As shown in the previous experiments, using additional information as extra coordinates does increases the accuracy. In the *third* configuration, we also use all additional information as extra coordinates, and we add sulcal depth information on graph nodes ($G = D^{-1}\exp(S)^{-1}$ in (2)) and on graph edges (in (1), $w_{ij} = 1/(d(i,j)+\epsilon)\exp(-(s_i - s_j)^2/2\sigma_s^2)$ where σ_s is a regularization term). This configuration is actually the best one in which our method performed. The average overlap ratio is 66% (or 88% of FreeSurfer's performance). This suggests that

(*G precentral*), 41 (*Medial wall*), 42 (*Pole occipital*), 45 (*S central*), 47 (*S cingulate Main part and Intracingulate*), 59 (*S intraparietal and Parietal transverse*), 80 (*S temporal superior*).

sulcal depth provides crucial information when matching brain hemispheres, as has been previously suggested by Fischl *et al.* [6]. The *fourth* configuration adds cortical thickness as additional information on graph nodes ($G = D^{-1}\exp(T)^{-1}$ in (2)) and on graph edges (in (1), $w_{ij} = 1/(d(i,j)+\epsilon)\exp(-(t_i - t_j)^2/2\sigma_t^2)$). Using cortical thickness alone actually worsen the overlap ratio to 58% (or 78% of FreeSurfer's performance). This suggests that cortical thickness may be a contradictory information in our spectral correspondence method. The *fifth* configuration uses cortical curvature as additional information on graph nodes ($G = D^{-1}\exp(K)^{-1}$ in (2)) and on graph edges (in (1), $w_{ij} = 1/(d(i,j)+\epsilon)\exp(-(\kappa_i - \kappa_j)^2/2\sigma_\kappa^2)$). Cortical curvature shows to be also a significant additional information as it increases the average overlap ratio to 64% (or 87% of FreeSurfer's performance). It is important to note that there is no perfect configuration of additional information. Our experiment showed that certain configurations perform better on particular parcellations rather than on others. The right configuration of additional information thus depends on which sulcal region of the brain should be matched. That said, our experiment suggests that sulcal depth and cortical curvature are significant additional information that improve our matching method.

4 Conclusion

Cerebral cortex matching is an important topic that facilitates basic computational study in neuroscience. Current, surface-based matching methods can be quite accurate, but very slow. We have proposed a new cortex matching algorithm based on spectral correspondence operating at speeds of several orders of magnitude faster than current methods. Furthermore, we extended spectral methods in order to use additional information as weights in graph nodes and as extra embedded coordinates with little or no computational expense. This added flexibility makes our spectral correspondence method a good candidate for brain studies involving many additional information. Our current best configuration of additional information were found when using sulcal depth, surface curvature, and cortical thickness, as extra embedded coordinates and sulcal depth on graph nodes and graph edges. Our brain surface matching method far outperforms the accuracy of the more commonly used volumetric methods and approaches FreeSurfer's level of accuracy when aligning sulcal regions (88% of FreeSurfer's performance). The vast increase in speed and the added flexibility when using additional information gives new perspectives to previously computationally prohibitive experiments. The contribution new features incorporated to help improve the matching (e.g., anatomical or functional features extracted from various data sources) can be tested. Quick parameter sweeps can be performed to isolate the best parameter value sets. These computationally intensive experiments can help us to understand what features are consistently correlated with brain areas across individuals and what their role are during the development of the cortical folding pattern. Currently, the correspondences found with the pairs of closest spectral neighbors. New schemes, such as the Relaxation Labeling as proposed in [23], will be tested and might improve accuracy. Future work will be to test different weighting functions (both based on nodes and edges), to incorporate more brain information (e.g., vascular density, MRI intensity), to evaluate the performance using additional cortical areas, and to test hypotheses about the relative importance of these features.

Acknowledgments

The authors would like to specially thank Gareth Funka-Lea and the financial support of the National Science and Environment Research Council (NSERC).

References

1. K. Amunts, A. Malikovic, H. Mohlberg, T. Schormann, and K. Zilles. Brodmann's areas 17 and 18 brought into stereotaxic space-where and how variable? *NeuroImage*, 11(1), 2000.
2. Q. Anqi, D. Bitouk, and M.I. Miller. Smooth functional and structural maps on the neocortex via orthonormal bases of the Laplace-Beltrami operator. *Trans. Med. Im.*, 25(10), 2006.
3. F. Chung. *Spectral Graph Theory (CBMS Conf. in Math., No. 92)*. AMS, 1997.
4. H. A. Drury, D. C. Van Essen, S. C. Joshi, and M. I. Miller. Analysis and comparison of areal partitioning schemes using 2-D fluid deformations. *NeuroImage*, 3, 1996.
5. B. Fischl, N. Rajendran, E. Busa, J. Augustinack, O. Hinds, B. T. Thomas Yeo, H. Mohlberg, K. Amunts, and K. Zilles. Cortical folding patterns and predicting cytoarchitecture. *Cereb Cortex*, 18(8), 2007.
6. B. Fischl, M. I. Sereno, R. B. Tootell, and A. M. Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human Brain Mapping*, 8(4), 1999.
7. B. Fischl, A. van der Kouwe, C. Destrieux, E. Halgren, F. Segonne, D. H. Salat, E. Busa, L. J. Seidman, J. Goldstein, D. Kennedy, V. Caviness, N. Makris, B. Rosen, and A. M. Dale. Automatically parcellating the human cerebral cortex. *Cereb. Cortex*, 14(1), 2004.
8. L. Grady and J. R. Polimeni. *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer, 2010.
9. O. P. Hinds, N. Rajendran, J. R. Polimeni, J. C. Augustinack, G. Wiggins, L. L. Wald, Diana H. Rosas, A. Potthast, E. L. Schwartz, and B. Fischl. Accurate prediction of V1 location from cortical folds in a surface coordinate system. *NeuroImage*, 39(4), 2008.
10. V. Jain and H. Zhang. Robust 3D shape correspondence in the spectral domain. In *Int. Conf. on Shape Mod. and App.*, 2006.
11. G. Lohmann, D. Y. von Cramon, and A. C. Colchester. Deep sulcal landmarks provide an organizing framework for human cortical folding. *Cereb Cortex*, 18(6), 2008.
12. D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer. Articulated shape matching using Laplacian eigenfunctions and unsupervised point registration. In *CVPR*, 2008.
13. A. Myronenko and X. Song. Point-set registration: Coherent point drift. *PAMI*, 2009.
14. M. Niethammer, M. Reuter, F.-E. Wolter, S. Bouix, N. Peinecke, M.-S. Koo, and M. Shenton. Global Medical Shape Analysis Using the Laplace-Beltrami Spectrum. In *MICCAI*, 2007.
15. M. Reuter. Hierarchical shape segmentation and registration via topological features of Laplace-Beltrami eigenfunctions. *Int. Journal Comp. Vis.*, 2009.
16. M. Reuter, F. E. Wolter, M. Shenton, and M. Niethammer. Laplace-Beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis. *Comput. Aided Des.*, 41(10), 2009.
17. G. L. Scott and H. C. Longuet-Higgins. An algorithm for associating the features of two images. *Proc Bio. Sc.*, 244(1309), 1991.
18. L. S. Shapiro and J. M. Brady. Feature-based correspondence: an eigenvector approach. *Image Vis. Comp.*, 10(5), 1992.
19. J. Talairach, G. Szikla, P. Tournoux, A. Prosalentis, M. Bordas-Ferrier, L. Covello, M. Iacob, and E. Mempel. *Atlas d'anatomie stereotaxique du telencephale*. Masson, Paris, 1967.
20. P. Thompson and A. W. Toga. A surface-based technique for warping three-dimensional images of the brain. *Trans. on Med. Im.*, 15(4), 1996.
21. S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *PAMI*, 10(5), 1988.
22. D. C. Van Essen and H. A. Drury. Structural and functional analyses of human cerebral cortex using a surface-based atlas. *J. Neurosci.*, 17(18), 1997.
23. Y. Zheng and D. Doermann. Robust point matching for nonrigid shapes by preserving local neighborhood structures. *PAMI*, 28(4), April 2006.