Proceedings of Machine Learning Research 1–12

GeoMorph: Geometric Deep Learning for Cortical Surface Registration

Mohamed A. Suliman*, **Logan Z. J. Williams**, **Abdulah Fawaz** and **Emma C. Robinson**

{MOHAMED.SULIMAN, LOGAN.WILLIAMS, ABDULAH.FAWAZ, EMMA.ROBINSON}@KCL.AC.UK

Department of Biomedical Engineering, School of Biomedical Engineering and Imaging Science, King's College London, London, SE1 7EH, UK

Abstract

We present GeoMorph, a geometric deep learning image registration framework that takes two cortical surfaces on the spherical space and learns a smooth displacement field that aligns the features on the moving surface to those on the target. GeoMorph starts with feature extraction: independently extracting low-dimensional feature representations for each input surface using graph convolutions. These learned features are then registered in a deep-discrete manner by learning the optimal displacement for a set of control points that optimizes the overlap between features across the two surfaces. To ensure a smooth deformation, we propose a regularization network that considers the input sphere structure based on a deep conditional random field (CRF), implemented using a recurrent neural network (RNN). Results show that GeoMorph improves over existing deep learning methods by improving alignment whilst generating smoother and more biologically plausible deformations. Performance is competitive with classical frameworks, generalizing well even for subjects with atypical folding patterns.

Keywords: Geometric deep learning, unsupervised learning, image registration.

1. Introduction

Cortical surface registration is an essential step in neuroimaging research that allows the comparison of common features of brain organization across subjects by mapping all data to a global average space in which these features overlap. Such alignment is generally driven using univariate summary measures of cortical folding, e.g., sulcal depth or average surface curvature (Fischl, 2012; Yeo et al., 2009; Robinson et al., 2014); in some cases, alignment of the cortical functional organization is also performed (Nenning et al., 2017; Robinson et al., 2014, 2018).

Classical approaches to cortical registration typically work following the projection of the cortical anatomy to a sphere; this simplifies the registration problem from one of matching complex shapes in 3D to one of matching functions on a smooth 2D surface. Registration is then performed by optimizing a similarity measure between the features on the target sphere and those on the deformed source sphere while enforcing smoothness constraints. Examples of such approaches include Freesurfer (Fischl et al., 1999), Spherical Demons (SD) (Yeo et al., 2009), and Multimodal Surface Matching (MSM) (Robinson et al., 2014, 2018). Such methods are meant to optimize a cost function for each pair of input images to improve feature overlap (hence, exhibit long execution times) while constraining the

* Corresponding author.

solution to be as smooth (or biologically plausible) as possible. However, they differ in calculating similarity, imposing regularization and diffeomorphism[1], and optimizing the similarity cost. Discrete-based optimization registration methods (Robinson et al., 2014, 2018) have shown advantages in learning functional mappings and large deformations over classical-based optimization methods. Note that while the existence of a diffeomorphism has been considered a prerequisite for cortical surface registration, emerging evidence shows that cortical topology can vary in a way that breaks this assumption (Glasser et al., 2016).

Recently, deep learning registration methods (Balakrishnan et al., 2019; Dalca et al., 2019; De Vos et al., 2019; Heinrich, 2019; Pielawski et al., 2020; Shao et al., 2021) have gained considerable interest on the grounds that they have faster execution times, are better at learning the space of variation to tackle topographical variation, and are more efficient in learning population-specific templates. Whilst originally developed for 2D Euclidean domains, increasing efforts have been made to extend these methods to surfaces, including S3Reg (Zhao et al., 2021), which learns displacements via the implementation of a spherical U-net network (Zhao et al., 2019), which compares the overlap of moving and target features at baseline, and seeks to enforce diffeomorphisms using the scaling and squaring approach of (Dalca et al., 2019). A fundamental limitation of S3Reg is that the hexagonal filter implemented in (Zhao et al., 2019) is not rotationally equivariant due to the lack of the global spherical coordinate; hence, it flips directions at the poles and generates distortions. S3Reg overcomes this by using a combination of three networks, each trained on a different rotated version of the input. In contrast, DDR (Suliman et al., 2022) addresses the problem of rotational equivariance by employing MoNet convolutions (Monti et al., 2017) (learned from a mixture of Gaussian kernels) to learn deformations in spherical space. Recent work (Fawaz et al., 2021) showed that MoNet convolutions could be trained to be rotationally equivariant. Unlike S3Reg, which learns deformations in a continuous setting, DDR learns deformations in a deep-discrete setting and is shown to improve alignment, especially in brains with atypical topographies, and provide smoother deformations (Suliman et al., 2022).

Contributions: Building on our recent work DDR (Suliman et al., 2022), we propose GeoMorph, a geometric deep learning framework for cortical surface registration based on MoNet that learns deformation in a discrete manner. GeoMorph is inspired by the deep-discrete registration framework (Heinrich, 2019), which shows capability in learning larger deformations than continuous-based ones. A key difference between DDR and GeoMorph is that while the registration in DDR is derived directly using input features, GeoMorph imposes a feature extraction network to learn a low-dimensional feature representation of the input. We hypothesize that such a step will improve cortical alignment by allowing the registration network to weight features based on their importance. This gains higher importance in multi-modal registration, where individual channels' contribution to the overall alignment becomes challenging. Moreover, we propose a new geometric-based regularization in GeoMorph that restricts the network to generate feasible deformation updates at training time in a way that minimizes the local mesh distortion, a constraint that is not present in DDR. We show that these contributions allow GeoMorph to surpass DDR in registration alignment quality and distortion measures.

1. A diffeomorphism is a globally one-to-one smooth and continuous mapping with derivatives that are invertible (i.e., non-zero Jacobian determinant) (Ashburner, 2007).

2. Method

Let $\mathbf{M}, \mathbf{F}$ be the vertices of the triangular meshes on the moving and fixed images, respectively, formed on a sphere $\mathcal{S}^2$ centred on the origin; each has N_d vertices, i.e., $\mathbf{M}, \mathbf{F} \in \mathbb{R}^{N_d \times 3}$. Our goal is to learn a spatial transformation $\mathbf{\Phi} : \mathbf{M} \rightarrow \mathbf{F}$ in the form

$$\mathbf{\Phi} = \mathcal{F}_{\boldsymbol{\eta}} \left(\mathbf{M}, \mathbf{F} \right) \tag{1}$$

that aligns the cortical features on $\mathbf{M}$ to those on $\mathbf{F}$ upon optimizing a dissimilarity metric $\mathcal{L}$ between them[2]. Such a problem is a severely ill-posed problem that has many possible solutions. Here, $\mathbf{\Phi}$ shifts the vertices of $\mathbf{M}$ to a new location such that the features on the newly deformed mesh overlap with those on $\mathbf{F}$. The function $\mathcal{F}_{\boldsymbol{\eta}}$ is obtained by a geometric deep neural network, with $\boldsymbol{\eta}$ being a set of learnable parameters.

Let $\{\mathbf{c}_i\}_{i=1}^{N_c} \in \mathbb{R}^{N_c \times 3}$ be a set of N_c control points on the moving sphere, generated from the vertices of a low-resolution icosphere (with $N_c << N_d$), and let $\{\mathbf{l}_i\}_{i=1}^{N_l} \in \mathbb{R}^{N_l \times 3}$ represent a set of label points, defined around each control point $\mathbf{c}_i$, that represent all potential endpoints of the transformation $\mathcal{F}_{\boldsymbol{\eta}}(\mathbf{c}_i)$. In all instances, the target labels are derived from the vertices of a higher-resolution icosphere (Fig. 2b). The objective of GeoMorph is, therefore, to learn in one go the optimal label assignment for each $\mathbf{c}_i$, such that the features on $\mathbf{M}$ and $\mathbf{F}$ are optimally aligned.

The general architecture of the GeoMorph network is shown in Fig. 1 with three parts: 1) feature extraction, which learns low-dimensional representations of the features on $\mathbf{M}$ and $\mathbf{F}$, 2) classifier network that estimates probabilities for each label assignment, i.e., $\mathbf{Q} = \text{Softmax}\left(\mathbf{U}\right) \in \mathbb{R}^{N_c \times N_l}$; 3) a CFR-RNN network which imposes smoothness by encouraging neighboring control points to take similar/adjacent labels.

2.1. Surface Convolutions and Pooling

Geometric convolutions: Surface convolutions are implemented using MoNet-style Gaussian mixture model convolutions (Monti et al., 2017). Let x be a vertex on a graph/surface and $y \in \mathcal{N}\left(x\right)$ be the points on the neighborhood of x each associated with a d-dimensional vector of pseudo-coordinates $\mathbf{u}\left(x, y\right)$. Then, MoNet convolutions are defined as

$$\left(f \star g\right)\left(x\right) = \sum_{j=1}^{J} g_j D_j\left(x\right) f, \tag{2}$$

where f is the input feature, g is a learnable filter, J is the dimensionality of the extracted patch, and $D\left(x\right) f$ is a patch operator given by $D_j\left(x\right) f = \sum_{y \in \mathcal{N}(x)} w_j\left(\mathbf{u}\left(x, y\right)\right) f\left(y\right), \forall j$ which extracts the values of f from the surface and then maps it at the neighborhood of x using learnable filter weights w_j. The weights w_j are formulated using the Gaussian function, i.e., $w_j\left(\mathbf{u}\right) = \exp\left(-\frac{1}{2}\left(\mathbf{u} - \boldsymbol{\mu}_j\right)^T \mathbf{\Sigma}_j^{-1}\left(\mathbf{u} - \boldsymbol{\mu}_j\right)\right)$, where $\mathbf{\Sigma}_j$ and $\boldsymbol{\mu}_j$ are learnable matrix and mean vector of the Gaussian kernel, respectively.

Icosphere resolutions: Starting from an icosphere of order 0, which has 20 faces, 30 edges, and 12 vertices, higher order resolutions can be generated by hierarchically adding a

2. We assume that $\mathcal{S}^2$ is parametrized by different resolutions (orders) of regularly sampled icospheres.

new vertex to the center of each edge in each triangle (see Fig. 2a). Let the number of the vertices at the current resolution level be N, then, the next higher resolution level will have $(N \times 4) - 6$ vertices, while the previous lower resolution level will have $(N + 6)/4$ vertices.

Surface downsampling/upsampling: Based on the icosphere nature above, we define the downsampling as the process of extracting only the vertices of the lower icosphere order. For the upsampling, we add new vertices as the average of their direct neighbors (Fig. 2a).

Surface pooling: We define the pooling operation as the process of replacing the vertex and its neighbors by the mean or the max of the accumulated features from all of them. Hence, we obtain a downsampled icosphere $N \rightarrow (N + 6)/4$ with new features.

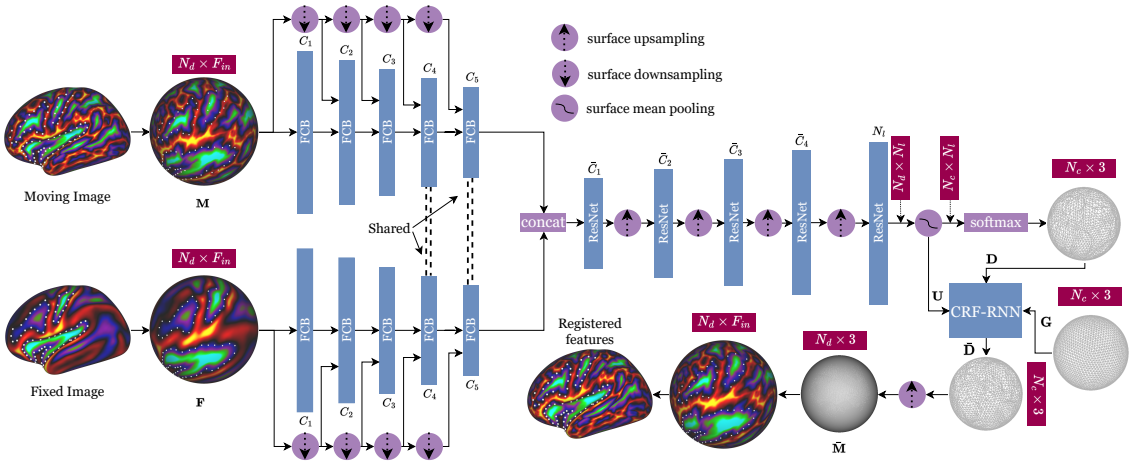


Figure 1: GeoMorph network architecture. The dimensions in red boxes shows the input and the output dimensions at different network stages.

Figure 2: a) Up and downsampling on icospheres. b) Example of a control point with its labels on the surface. c) FCB architecture. d) ResNet architecture.

2.2. Feature Extraction

The network takes the input features and the mesh topology in the form of vertex locations and neighborhood structure $\mathcal{N}$, where $(i,j) \in \mathcal{N} \subset \{1,\ldots,N_d\}^2$ indicates that vertex i is connected to j by a triangle edge. A series of feature convolutional blocks (FCBs) are applied to learn a low-dimensional feature space. The features on $\mathbf{M}$ and $\mathbf{F}$ are learned independently on separate paths, with only the weights of the last two FCBs being shared.

At each FCB stage i, the number of vertices V_i is related to the previous one through $V_i = (V_{i-1}+6)/4$, with $V_1 = V_{\mathbf{M}}$. A total of C_i features are learned at each stage i using two MoNet convolutional filters with a kernel size of 3, spherical polar pseudo-coordinates, and mean aggregation operators followed by a LeakyReLU activation with parameter 0.2. The output features are then passed through a surface max pooling operator. To allow for global feature incorporation, the max pooling output is concatenated with a downsampled version of the LeakyReLU output. The result is then passed through a gate function G with $G = A$ for $i = 1,\ldots,4$ and $G = B$ for $i = 5$ (see Fig. 2c).

2.3. Classifier Network

The learned features from the previous stage are concatenated and passed through a series of five ResNet-inspired blocks, each learning $\bar{C}_i$ features, with the last one learning N_l features. The output of each network is first upsampled to the next icosphere order and then passed to the next stage. At each block, we perform two surface convolutions followed by a LeakyReLU activation with parameter 0.2 (see Fig. 2d). The output of the final ResNet, which is of dimension $N_d \times N_l$, is downsampled to the desired control grid resolution, i.e., $\mathbf{U} \in \mathbf{R}^{N_c \times N_l}$. The optimal label assignment is obtained as a softmax operation on $\mathbf{U}$. Finally, we deform $\mathbf{c}_i$ using the spherical coordinates of the labels.

2.4. Regularization Network

As the resulting deformation does not incorporate any constraint, we introduce the CRF-RNN network (Zheng et al., 2015) to impose smoothness upon forcing neighboring control points to deform into adjacent/similar label points. Consider the energy cost function

$$E = \sum_i Q_{(\mathbf{c}_i,\mathbf{l}_i)} + \sum_{i \neq j} \varphi\left(\mathbf{l}_{\mathbf{c}_i}, \mathbf{l}_{\mathbf{c}_j}\right), \tag{3}$$

where $Q_{(\mathbf{c}_i,\mathbf{l}_i)}$ is the likelihood of deforming $\mathbf{c}_i$ to $\mathbf{l}_i$ while $\varphi\left(\mathbf{l}_{\mathbf{c}_i}, \mathbf{l}_{\mathbf{c}_j}\right) = \mu\left(\mathbf{l}_i, \mathbf{l}_j\right) K_G\left(\mathbf{l}_{\mathbf{c}_i}, \mathbf{l}_{\mathbf{c}_j}\right)$ is the cost of deforming $\mathbf{c}_i$ and $\mathbf{c}_j$ to $\mathbf{l}_i$ and $\mathbf{l}_j$, respectively. Here, μ is a learnable label compatibility function that captures correspondences between different pairs of label points, while K_G is a Gaussian kernel (Krähenbühl and Koltun, 2011; Zheng et al., 2015) of the form

$$K_G\left(\mathbf{l}_{\mathbf{c}_i}, \mathbf{l}_{\mathbf{c}_j}\right) = \omega\left(\mathbf{c}_i, \mathbf{c}_j\right) \exp\left(-\frac{1}{2\gamma^2}\left(\mathbf{l}_{\mathbf{c}_i} - \mathbf{l}_{\mathbf{c}_j}\right)^T \mathbf{\Lambda}\left(\mathbf{l}_{\mathbf{c}_i} - \mathbf{l}_{\mathbf{c}_j}\right)\right). \tag{4}$$

ω are learnable filter weights, γ is a kernel parameter, $\mathbf{l}_{\mathbf{c}_i}$ is the new spatial location of the deformed $\mathbf{c}_i$, while $\mathbf{\Lambda}$ is a symmetric, positive-definite, kernel characterization matrix. The function in (3) is optimized using the Recurrent Neural Network (RNN) implementation of (Zheng et al., 2015), which is based on learning multiple iterations of a mean-field CRF.

2.5. Optimization

The deformed control grid from the CRF-RNN network $\bar{\mathbf{D}} \in \mathbb{R}^{N_c \times 3}$ is upsampled to the input level $\bar{\mathbf{M}} \in \mathbb{R}^{N_d \times 3}$ using bilinear interpolation. Then, we resample the moving image features to $\bar{\mathbf{M}}$ using bilinear interpolation and compare with the fixed image features.

Loss functions: The network optimization is derived using an unsupervised loss function $\mathcal{L}$ in the form:

$$\mathcal{L} = \lambda_{\text{sim}}\mathcal{L}_{\text{sim}} + \lambda_{\text{sm}}\mathcal{L}_{\text{sm}} + \lambda_{\text{arap}}\mathcal{L}_{\text{arap}}. \tag{5}$$

The term $\mathcal{L}_{\text{sim}}$ measures the similarity between the features on $\mathbf{F}$ and those on $\bar{\mathbf{M}}$. We use a measure that is a sum of the MSE and cross-correlation (CC), i.e.,

$$\mathcal{L}_{\text{sim}} = \frac{1}{N_d}\sum_{i=1}^{N_d}\left(\left|\left|\mathbf{F}_{\mathbf{v}_i} - \bar{\mathbf{M}}_{\mathbf{v}_i}\right|\right|_2^2 - \frac{cov\left(\mathbf{F}_{\mathbf{v}_i}, \bar{\mathbf{M}}_{\mathbf{v}_i}\right)}{\sigma_{\mathbf{F}_{\mathbf{v}_i}}\sigma_{\bar{\mathbf{M}}_{\mathbf{v}_i}}}\right), \tag{6}$$

where $\mathbf{F}_{\mathbf{v}_i}, \bar{\mathbf{M}}_{\mathbf{v}_i}$ denote the corresponding features at vertex i, $cov\left(\cdot,\cdot\right)$ is the covariance operator, and σ is the standard deviation. $\mathcal{L}_{\text{sm}}$ is introduced to provide user control over the balance between accurate alignment and smooth deformation and is formulated as a diffusion regularization penalty on the gradients of the $\mathbf{\Phi}$, i.e., $\mathcal{L}_{\text{sm}} = (|\bigtriangledown\mathbf{\Phi}_{\mathbf{x}}| + |\bigtriangledown\mathbf{\Phi}_{\mathbf{y}}| + |\bigtriangledown\mathbf{\Phi}_{\mathbf{z}}|)$, where $\mathbf{x}, \mathbf{y}, \mathbf{z}$ refer to the cardinal directions. We apply the hexagonal filter in (Zhao et al., 2019) to compute $\bigtriangledown$.

Finally, the *as-rigid-as-possible* loss $\mathcal{L}_{\text{arap}}$ is introduced to regularize the trajectory of deformations updates during training. Let $\bar{\mathbf{D}}_k$ and $\bar{\mathbf{D}}_{k+1}$ be the outputs of the CRF-RNN at time k and $k+1$, respectively, with $\bar{\mathbf{d}}_{i,k}$ being vertex i at time k. $\mathcal{L}_{\text{arap}}$ locally preserves the shape of the mesh as we go from $\bar{\mathbf{D}}_k$ to $\bar{\mathbf{D}}_{k+1}$ by restricting the mesh to change as rigidly as possible. To formulate $\mathcal{L}_{\text{arap}}$, we use the as rigid as possible measure (Sorkine and Alexa, 2007) as

$$\mathcal{L}_{\text{arap}}\left(\bar{\mathbf{D}}_k, \bar{\mathbf{D}}_{k+1}\right) = \frac{1}{2}\min_{\substack{\mathbf{R}_i \in SO(3)\\ i=1,\ldots,N_c}} \sum_{i,j\in\mathcal{N}} \left|\left|\mathbf{R}_i\left(\bar{\mathbf{d}}_{j,k} - \bar{\mathbf{d}}_{i,k}\right) - \left(\bar{\mathbf{d}}_{j,k+1} - \bar{\mathbf{d}}_{i,k+1}\right)\right|\right|_2^2. \tag{7}$$

As equation (7) shows, $\mathcal{L}_{\text{arap}}$ rotates $\bar{\mathbf{d}}_{i,k}$ to the corresponding $\bar{\mathbf{d}}_{i,k+1}$ and then regularizes the deviation from the local rigid transformation (Eisenberger et al., 2021). The rotation matrices $\mathbf{R}_i$ can be computed in a closed form, as shown in (Sorkine and Alexa, 2007).

Coarse-to-fine-registration: As with surface registration methods, we perform multi-stage registration in the form of coarse-to-fine using two networks. The first network is trained to align features using a coarse grid of control points. The result is upsampled to the next network level and passed to the second network that uses a higher resolution control grid with higher label resolution.

3. Experiments Setup

We conduct a series of experiments to register multiple cortical surfaces to a global template.

Dataset: We use the adult Human Connectome Project (HCP) dataset (Glasser et al., 2013), which consists of cortical surface metrics and meshes. Using barycentric interpolation, we resample left cortical hemispheres to an icosphere of order 6. A total of 1110 cortical

surfaces were used in the experiments with a split of 888-111-111 train-validation-test. To simplify the problem, we drive registration using sulcal depth as the sole feature.

Training: The network is trained by optimizing (5) using ADAM (Kingma and Ba, 2014) with a learning rate of 2×10^{-4}. MoNet filters are implemented using the PyTorch Geometric library (Fey and Lenssen, 2019). During the coarse registration stage, we set the control grid to be the vertices of an icosphere of order 2, i.e., $N_c = 126$, and we set $N_l = 600$ lying on an icosphere of order 5. At the fine stage, we raise the control grid icosphere resolution to order 4, i.e., $N_c = 2542$, and we set $N_l = 1000$, increasing its resolution to icosphere of order 8. The features of the FCB are set to be $\{C_i\}_{i=1}^{5} = \{32, 32, 64, 64, 128\}$, during the coarse and the fine stages. For the ResNet, we use $\{\bar{C}_i\}_{i=1}^{4} = \{256, 128, 64, 64\}$ during the coarse stage, and $\{\bar{C}_i\}_{i=1}^{4} = \{256, 128, 128, 128\}$ during the fine stage. Finally, we set $\lambda_{\text{sim}} = 1$, $\lambda_{\text{sm}} = 1.5$, and $\lambda_{\text{arap}} = 0.02$ during the coarse stage, and $\lambda_{\text{sim}} = 1$, $\lambda_{\text{sm}} = 0.5$, and $\lambda_{\text{arap}} = 0.02$ during the fine stage. The network is trained over 150 epochs (found to be enough to reach convergence), and we report the performance with the best similarity score on the validation set.

Benchmarks: We benchmark against SD, MSM, Freesurfer, and the learning-based methods of S3Reg and DDR. We validate against the official implementations of SD[3], MSM Pair, MSM Strain[4], S3Reg[5], and DDR[6]. Parameters for all competing methods were optimised in (Suliman et al., 2022). Note that all these frameworks, except DDR, register two surfaces at 4 levels of icosphere subdivisions (coarse to fine). DDR registration is done using two icosphere levels.

Evaluation metrics: We evaluate our performance using CC between the features on the deformed mesh and the fixed mesh, and the distortion measure, which signifies how much individual triangles on the moving mesh are distorted during registration. We measure the areal strain (J) and the shape strain (R). Here, $J = \lambda_1\lambda_2$ and $R = \lambda_1/\lambda_2$, where λ_1 and λ_2 represent the eigenvalues of the local deformation gradient F_{pqr} estimated from the deformation of each triangular face, defined by vertices: p, q, r. Note that $\log_2 J$ is equivalent to areal distortion, and $\log_2 R$ is equivalent to shape distortion.

4. Results

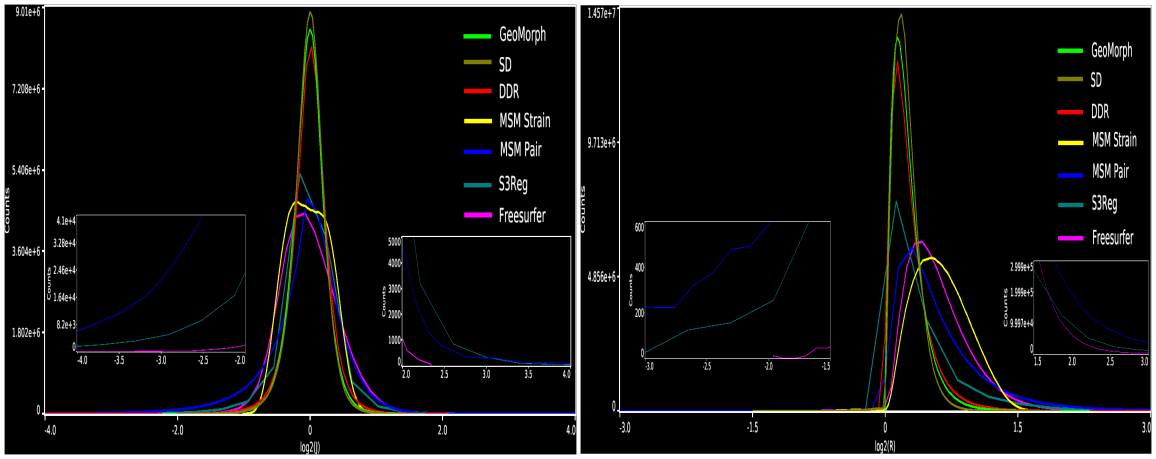
To provide a fair comparison between different methods, we fix the CC of all algorithms by using the parameters that give the chosen CC level for each method and compare other performance measures. From the various runs of the benchmark registration methods, we found that all algorithms provide a CC of approximately 0.88; hence, we use it as our point of reference. In Table. 1, we report the exact mean CC across all runs as well as the mean, max, 95th, and 98th percentiles of the areal and the shape distortions for all algorithms. The table shows that GeoMorph performs closely to SD, MSM Strain, and DDR, while S3Reg and MSM Pair provide the worst performance. The result also shows improvements in the performance of GeoMorph compared to DDR in all distortions measures. GeoMorph, however, has a higher runtime than DDR. The time is measured on a PC with NVIDIA Titan RTX 24GB GPU and Intel Core i9-9820X 3.30 GHz CPU.

3. https://github.com/ThomasYeoLab/CBIG 4. https://github.com/ecr05/MSM_HOCR
5. https://github.com/zhaofenqiang/SphericalUNetPackage
6. https://github.com/mohamedasuliman/DDR

Table 1: Distortions measures and average runtime for different methods at CC $\sim$ 0.88. Classical methods (top) and learning-based methods (bottom).

Methods	CC Similarity	Areal Distortion				Shape Distortion				Avg. Time	
		Mean	Max	95%	98%	Mean	Max	95%	98%	CPU	GPU
Freesurfer	0.75	0.34	11.73	0.82	1.00	0.63	6.77	1.29	1.54	30 min	-
MSM Pair	0.877	0.41	9.17	1.24	1.76	0.62	9.05	1.61	2.16	13 min	-
MSM Strain	0.880	0.27	1.06	0.53	0.66	0.64	1.93	1.17	1.30	1 hour	-
SD	0.875	0.18	2.00	0.50	0.65	0.24	1.98	0.50	0.65	1 min	-
S3Reg	0.875	0.26	22.22	0.82	1.16	0.51	21.65	1.35	2.0	8.8 sec	8.0 sec
DDR	0.875	**0.19**	2.66	0.53	0.71	**0.26**	3.14	0.66	0.86	**7.7 sec**	**2.3 sec**
GeoMorph	0.875	**0.19**	**2.21**	**0.52**	**0.65**	**0.26**	**2.30**	**0.62**	**0.78**	18 sec	3.2 sec

In Fig. 3, we plot the full histogram of the areal and the shape distortions for all test subjects. Based on Fig. 3a, we observe that most of the areal distortions of SD, GeoMorph, and DDR are around zero, with GeoMorph having more distortions close to zero than DDR. MSM Pair, S3Reg, and Freesurfer (with a lower degree) all have extreme distortions across subjects, as presented by long tails. The same trend appears in Fig. 3b, where we see that most SD, GeoMorph, and DDR shape distortions centered around one, with other methods having extreme distortions.

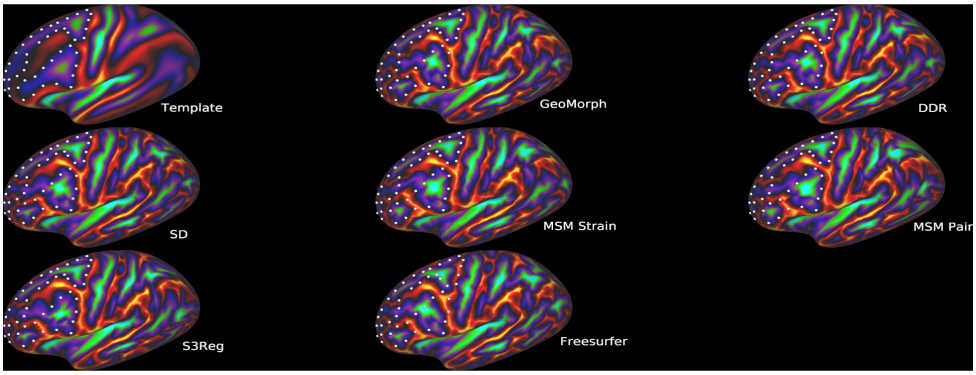


(a) Areal Distortion. (b) Shape Distortion.

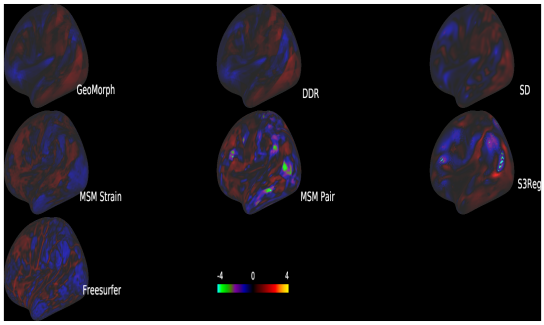
Figure 3: Histogram plots comparing areal and shape distortions across all test subjects.

In Fig. 4, we evaluate the methods' alignments quality on a subject with atypical cortical folding patterns and also provide the areal and the shape distortions of the methods. Fig. 4 shows that GeoMorph, DDR, SD, and MSM Strain all provide good alignment with minimal distortions. In contrast, the alignments by MSM Pair and S3Reg are highly distorted.

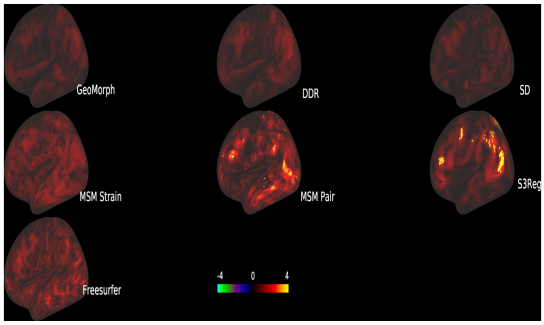
In Fig. 5, we provide a close comparison between GeoMorph, DDR, SD, and MSM Strain for the results in Fig. 4 by dropping the threshold to a smaller value. The figure clearly



(*a*) Alignment Quality.



(*b*) Areal Distortion.



(*c*) Shape Distortion.

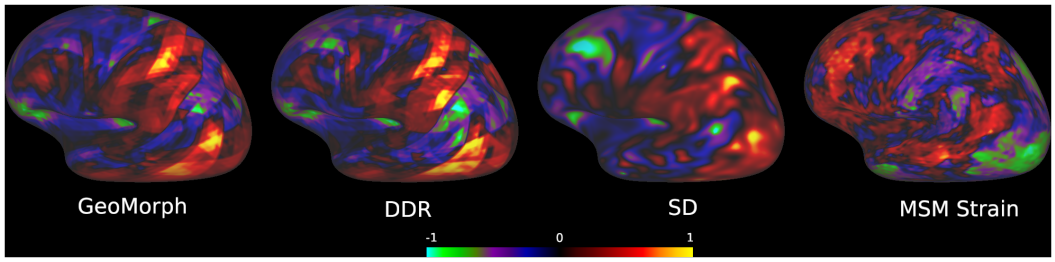
Figure 4: a) Methods performance on a subject with atypical cortical folding patterns. b) Associated areal distortions. c) Associated shape distortions.

shows the improvement in distortion in GeoMorph over DDR in both measures. Fig. 5 also indicates that in terms of areal distortion, GeoMorph has more distortions close to zero with more positive and less negative distortions than MSM Strain. GeoMorph also has an overall better shape distortion than MSM Strain.

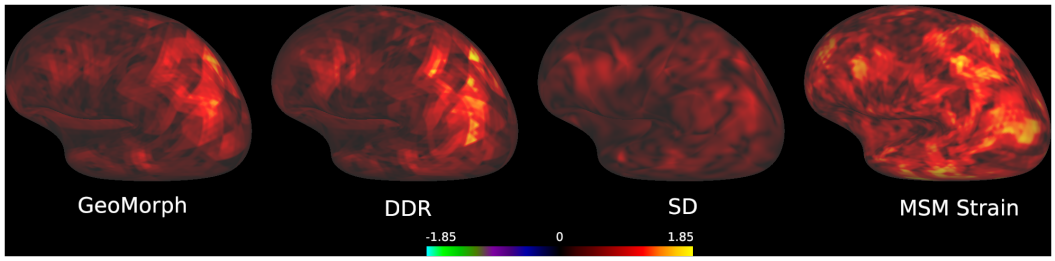
Finally, we provide in Table. 2 the dice overlap of all methods computed as average overall test subjects along with the 98% of the areal distortions all at CC ~ 0.88. Each test subject's sulcal depth map is binarized with sulci set to 1 and everything else set to 0. Table. 2 shows that in terms of dice overlap and distortion, GeoMorph performance is comparable with SD and MSM Strain and is better than all other methods. MSM Pair, which has a slightly better dice overlap than GeoMorph, exhibits an approximately 3 times worse distortion.

Table 2: Dice overlap and the 98% of areal distortions for different methods at CC ~ 0.88.

	Freesurfer	**MSM Pair**	**MSM Strain**	**SD**	**S3Reg**	**DDR**	**GeoMorph**
Dice	0.779	**0.857**	**0.856**	0.849	0.849	0.848	0.848
Dist 98%	1.00	1.76	0.66	**0.65**	1.16	0.71	**0.65**



(a) Areal Distortion.



(b) Shape Distortion.

Figure 5: Close performance comparison between distortions in Fig. 4.

5. Conclusions

In this work, we developed a geometric deep learning image registration framework (GeoMorph), based on the deep-discrete registration methods, that extracts low dimensional feature representations on the input features and then learns deformations as a multi-label classification problem, conditioned by a CRF to prevent neighboring points from deforming in entirely different directions. Moreover, we propose applying a geometric-based regularization penalty during training time to minimize mesh distortions. Results show that distortions obtained through GeoMorph outperform other deep-learning-based methods and are very close to those obtained by classical-based methods. The structure of GeoMorph is expected to allow for more efficient handling of multi-modal alignment, which will be the future focus of this work.

References

John Ashburner. A fast diffeomorphic image registration algorithm. *Neuroimage*, 38(1): 95–113, 2007.

Guha Balakrishnan, Amy Zhao, Mert R Sabuncu, John Guttag, and Adrian V Dalca. Voxelmorph: a learning framework for deformable medical image registration. *IEEE transactions on medical imaging*, 38(8):1788–1800, 2019.

Adrian Dalca, Marianne Rakic, John Guttag, and Mert Sabuncu. Learning conditional deformable templates with convolutional networks. *Advances in neural information pro-*

cessing systems, 32, 2019.

Bob D De Vos, Floris F Berendsen, Max A Viergever, Hessam Sokooti, Marius Staring, and Ivana Išgum. A deep learning framework for unsupervised affine and deformable image registration. *Medical image analysis*, 52:128–143, 2019.

Marvin Eisenberger, David Novotny, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. Neuromorph: Unsupervised shape interpolation and correspondence in one go. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7473–7483, 2021.

Abdulah Fawaz, Logan ZJ Williams, Amir Alansary, Cher Bass, Karthik Gopinath, Mariana da Silva, Simon Dahan, Chris Adamson, Bonnie Alexander, Deanne Thompson, et al. Benchmarking geometric deep learning for cortical segmentation and neurodevelopmental phenotype prediction. *bioRxiv*, 2021.

Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

Bruce Fischl, Martin I Sereno, Roger BH Tootell, and Anders M Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human brain mapping*, 8(4):272–284, 1999.

Matthew F Glasser, Stamatios N Sotiropoulos, J Anthony Wilson, Timothy S Coalson, Bruce Fischl, Jesper L Andersson, Junqian Xu, Saad Jbabdi, Matthew Webster, Jonathan R Polimeni, et al. The minimal preprocessing pipelines for the human connectome project. *Neuroimage*, 80:105–124, 2013.

Matthew F Glasser, Timothy S Coalson, Emma C Robinson, Carl D Hacker, John Harwell, Essa Yacoub, Kamil Ugurbil, Jesper Andersson, Christian F Beckmann, Mark Jenkinson, et al. A multi-modal parcellation of human cerebral cortex. *Nature*, 536(7615):171–178, 2016.

Mattias P Heinrich. Closing the gap between deep and conventional image registration using probabilistic dense displacement networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 50–58. Springer, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011.

Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5115–5124, 2017.

Karl-Heinz Nenning, Hesheng Liu, Satrajit S Ghosh, Mert R Sabuncu, Ernst Schwartz, and Georg Langs. Diffeomorphic functional brain surface alignment: Functional demons. *NeuroImage*, 156:456–465, 2017.

Nicolas Pielawski, Elisabeth Wetzer, Johan Öfverstedt, Jiahao Lu, Carolina Wählby, Joakim Lindblad, and Natasa Sladoje. Comir: Contrastive multimodal image representation for registration. *Advances in neural information processing systems*, 33:18433–18444, 2020.

Emma C Robinson, Saad Jbabdi, Matthew F Glasser, Jesper Andersson, Gregory C Burgess, Michael P Harms, Stephen M Smith, David C Van Essen, and Mark Jenkinson. Msm: a new flexible framework for multimodal surface matching. *Neuroimage*, 100:414–426, 2014.

Emma C Robinson, Kara Garcia, Matthew F Glasser, Zhengdao Chen, Timothy S Coalson, Antonios Makropoulos, Jelena Bozek, Robert Wright, Andreas Schuh, Matthew Webster, et al. Multimodal surface matching with higher-order smoothness constraints. *Neuroimage*, 167:453–465, 2018.

Wei Shao, Linda Banh, Christian A Kunder, Richard E Fan, Simon JC Soerensen, Jeffrey B Wang, Nikola C Teslovich, Nikhil Madhuripan, Anugayathri Jawahar, Pejman Ghanouni, et al. Prosregnet: A deep learning framework for registration of mri and histopathology images of the prostate. *Medical image analysis*, 68:101919, 2021.

Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007.

Mohamed A Suliman, Logan ZJ Williams, Abdulah Fawaz, and Emma C Robinson. A deep-discrete learning framework for spherical surface registration. *arXiv preprint arXiv:2203.12999*, 2022.

BT Thomas Yeo, Mert R Sabuncu, Tom Vercauteren, Nicholas Ayache, Bruce Fischl, and Polina Golland. Spherical demons: fast diffeomorphic landmark-free surface registration. *IEEE transactions on medical imaging*, 29(3):650–668, 2009.

Fenqiang Zhao, Shunren Xia, Zhengwang Wu, Dingna Duan, Li Wang, Weili Lin, John H Gilmore, Dinggang Shen, and Gang Li. Spherical u-net on cortical surfaces: methods and applications. In *International Conference on Information Processing in Medical Imaging*, pages 855–866. Springer, 2019.

Fenqiang Zhao, Zhengwang Wu, Fan Wang, Weili Lin, Shunren Xia, Dinggang Shen, Li Wang, and Gang Li. S3reg: superfast spherical surface registration based on deep learning. *IEEE Transactions on Medical Imaging*, 40(8):1964–1976, 2021.

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.