Published in final edited form as:

Med Image Comput Comput Assist Interv. 2019 October ; 11766: 492–500.

doi:10.1007/978-3-030-32248-9_55.

Intrinsic Patch-Based Cortical Anatomical Parcellation Using Graph Convolutional Neural Network on Surface Manifold

Zhengwang Wu, Fenqiang Zhao, Jing Xia, Li Wang, Weili Lin, John H. Gilmore, Gang Li, Dinggang Shen

Department of Radiology and BRIC, University of North Carolina at Chapel Hill, Chapel Hill, USA

Abstract

Automatic parcellation of cortical surfaces into anatomically meaningful regions of interest (ROIs) is of great importance in brain analysis. Due to the complex shape of the convoluted cerebral cortex, conventional methods generally require three steps to obtain the parcellations. *First*, the original cortical surface is iteratively inflated and mapped onto a spherical surface with minimal metric distortion, for providing a simpler shape for analysis. *Then*, a registration or learning-based labeling method is adopted to parcellate ROIs on the mapped spherical surface. *Finally*, parcellation labels on the spherical surface are mapped back to the original cortical surface. Despite great success, spherical mapping of the original cortical surface is inherently sensitive to topological noise and cannot deal with the impaired brains that violate spherical topology. To address these issues, in this paper, we propose to directly parcellate the cerebral cortex on the original cortical surface manifold without requiring spherical mapping, by leveraging the strong learning ability of the graph convolutional neural networks. Also, we extend the convolution to the surface manifold using the kernel strategy, which enables us to overcome the notorious shape difference issue (e.g., different vertex number and connections) across different subjects. Our method aims to learn the highly nonlinear mapping between cortical attribute patterns (on local intrinsic surface patches) and parcellation labels. We have validated our method on a normal cortical surface dataset and a synthetic dataset with impaired brains, which shows that our method achieves comparable accuracy to the methods using spherical mapping, and works well on cortical surfaces violating the spherical topology.

Keywords

Cortical surface parcellation; Graph CNN

1 Introduction

The highly convoluted human cerebral cortex can be parcellated into neurobiologically meaningful regions [1], which are the foundation for many brain structural and functional studies [2]. E.g., the functional network studies need the parcellation ROIs to investigate the connections across different brain regions. Previously, many computational methods have been proposed [1,3–9] for automatic cortical surface parcellation. Typically, these methods

include three major steps. *First*, the original cortical surface is iteratively inflated and mapped onto a spherical surface with minimal metric distortion (in terms of edge lengths and triangle angles) [10]. *Then*, registration or learning based labeling methods are adopted to parcellate ROIs on the mapped spherical surface. *Finally*, the parcellation labels on the spherical surface are mapped back to the original cortical surface. The underlying rationale is that, compared to original cortical surface, the spherical surface is better amenable to surface registration and analysis. Despite its great success, a major drawback of this strategy is it cannot be applied to *either* the impaired cortex violating spherical topology *or* cortical surfaces with large topological noises. Because, in these two cases, the spherical mapping will have unavoidable large distortions. However, clinically, it is highly desired to know how impaired brains work.

To address this methodology gap, we propose a learning-based automatic parcellation method, which directly works on the original cortical surface manifold and thus is free of spherical mapping and registration. Our method learns the mapping between cortical attribute patterns on the original surface manifold and the parcellation labels, by leveraging the strong learning ability of the graph convolutional neural networks (GCNNs). The GCNNs can be regarded as an adaptation of the conventional CNNs from the Euclidean domain to the surface manifold. In the conventional CNN, the element of the convolution template (we use terminology template instead of kernel for distinguishing from Gaussian kernel function which we describe later) is a scalar value; while in GCNN, the convolution template element is a function, e.g., a Gaussian kernel function, whose parameters can be learned during the training stage. With this extension, our method can be applied to the surface manifold which often has inconsistent neighborhood definitions (such as different number of neighbors or connections). In the training stage, we feed the GCNN with intrinsic patches extracted from the original cortical surface manifold as well as their corresponding parcellation labels to train the model. Herein, we use the term intrinsic patches because (1) they are extracted based on the intrinsic metric, i.e., geodesic distance, and (2) they are charted based on the intrinsic local coordinate framework, whose principal axes are the minimal and maximal curvature directions. Then, in the testing stage, the extracted intrinsic patches and the trained models are applied for predicting the parcellation labels. We have validated our method on both normal cortical surfaces and synthesized impaired cortical surfaces. The experimental results show that our method achieves accuracy comparable to the conventional methods based on spherical mapping. Note that the main advantage of our method is that it can be naturally applied to impaired cortical surfaces which violate spherical topology. To the best of our knowledge, this is the first paper dealing with cortical parcellation on impaired cortical surfaces.

2 Method

Our parcellation method includes 4 components: (1) image preprocessing and cortical attributes computation; (2) informative intrinsic patch extraction; (3) GCNN training on intrinsic patches; and (4) network structure and postprocessing.

2.1 Image Preprocessing and Cortical Attribute Computation

The brain cortical surfaces in this study are reconstructed from neonatal brain MR images using an in-house computational pipeline, including skull-stripping, intensity inhomogeneity correction, tissue segmentation, and cortical surface reconstruction. Then, we further follow the well-accepted Freesurfer Desikan parcellation protocol [3] to manually parcellate the reconstructed cortical surfaces into different regions. The Freesurfer Desikan parcellation protocol partitions the cerebral cortex into 72 (36 for each hemisphere) regions based on the major cortical folds, which have been largely established at term birth. Hence, we apply it onto our neonate cortical surfaces. It is worth noting that the parcellation protocol provides us merely the training data, and other parcellations protocols can also be adopted.

To automatically learn how to parcellate the cerebral cortex, we further compute informative cortical attribute maps on cortical surfaces. These cortical attribute maps are used to characterize cerebral cortex in geometric viewpoints. Specifically, we compute the mean curvature, sulcal depth, shape index, and curvedness, because they provide geometric clues of the cortical folding patterns, which are the reference for manual parcellations. Note that all these cortical attributes can be computed on *either* normal *or* impaired cortical surfaces. In Fig. 1, we present these cortical attribute maps and their parcellation label map on a typical cortical surface.

2.2 Informative Intrinsic Patch Extraction

Formally, the cortical surface manifold is an undirected graph (V, E) , with V being the vertex set and E being the edge set. For any vertex on the convoluted cortical surface, we can extract a local intrinsic patch, which contains a set of vertices v_k^i whose geodesic distances to v_i are bounded by the predefined maximal geodesic distance ρ_{max} . We need to chart the vertices on the extracted patch with a reference coordinate framework for further computation. Herein, we use a local intrinsic coordinate framework, with its two principal directions defined by the minimal principal curvature and the maximal principal curvature directions. Figure 2(b) shows the minimal curvature direction field of a typical region. With the above method, given any vertex v_i and the predefined ρ_{max} , we can extract a local patch P_{v_i} . Any vertex in this patch is charted by two parameters (ρ, θ) , where ρ denotes geodesic distance from each vertex to the patch center, and θ denotes the angle between the shortest geodesic path and the minimal principal curvature direction, as illustrated in Fig. 2(c). We call this extracted patch as intrinsic patch, because *neither* the geodesic distances *nor* the charting parameters rely on how the cortical surface is embedded in the 3D space. This means the spatial relationship on these patches is independent of the pose and orientation of the brain. Of note, both the geodesic distance and the underlying shortest path between two vertices on the cortical surface can be computed efficiently through a wave-propagation [11].

Although we can extract a local intrinsic patch for each vertex, it does not necessarily mean that all the extracted patches are informative. Considering the prior knowledge that the parcellation boundaries of cortical surfaces generally appear at the sulcal fundi or gyral crests, we sample more patches from those locations, and meanwhile reduce number of patch samples for vertices sitting on sulcal walls. This can be achieved by thresholding the

mean curvature map of the cortical surface, since the sulcal fundi and gyral crests typically have large curvature magnitude. Figure 2(a) shows the typical sampling positions on a cortical surface.

2.3 Convolution on Graph

Convolutional neural networks (CNNs) are powerful methods for learning highly nonlinear mappings. A major barrier for using conventional CNNs on extracted intrinsic patches is that these patches are graphs, not images, which means that different patches contain different numbers of vertices and local connections. A straightforward solution is to reparametrize each local patch based on the current charting framework and interpolate at regular grid points to form a grid-like patch. However, it is difficult to manually select a reparameterization strategy to fit all patches for all cortical attributes, which might introduce unnecessary selection bias. To address this issue, instead of using the conventional CNN, we propose to use the graph CNN [12], which directly defines the convolutional operations on the graph and inherits the spatial invariance merit of the CNN. The GCNN is able to automatically learn the latent parametrization from the data, through which the convolution is extended to the graph.

Specifically, the GCNN first learns a patch operator, which uses several Gaussian kernels for latent reparameterization and interpolation in the patch. It is named *latent* because kernel parameters are learned during the network training, which enables a highly heterogeneous parametrization for the cortical attributes defined on the cortical surface manifold.

Specifically, for a local intrinsic patch P_{v_i} , we denote its vertices as $v_k^i = (\rho_k, \theta_k) \in P_{v_i}$, $k = 1, \dots, K_{v_i}$, where K_{v_i} is the vertex number of this patch, and (ρ_k, θ_k) are the coordinates of v_k^i based on the current charting framework. Then, regarding these vertices as control points, for the cortical attributes $\mathcal{F}$ defined on these control points, we can define a patch operator $\mathcal{D}_{\mathcal{F}}$, which uses Gaussian kernels to reparametrize the patch and can be formulated as:

$$\mathcal{D}_{\mathcal{F}}(v_k^i; \Theta) = \sum_j \mathcal{N}_j(\Theta_j) \mathcal{F}(v_k^i), \quad j = 1, \dots, J \quad (1)$$

where $\mathcal{N}_j(\Theta_j)$ denotes a Gaussian kernel with parameter Θ_j and $\Theta = \{\Theta_j; j = 1, \dots, J\}$. J is a hyper-parameter which defines the kernel number.

With the above patch operator, we have reparametrized the intrinsic patch in a uniform latent kernel space Θ , where the convolution operation is defined. The convolution for a patch vertex v_k^i with cortical attribute $\mathcal{F}(v_k^i)$ can be defined as a weighted sum over the responses of the kernels, i.e.,

$$\text{Conv}_{\mathcal{F}}(v_k^i) = \mathbf{w}(\Theta) \star \mathcal{D}_{\mathcal{F}}(v_k^i; \Theta) = \sum_{j=1}^J w_j(\Theta_j) \mathcal{N}_j(\Theta_j) \mathcal{F}(v_k^i) \quad (2)$$

Therefore, for our graph convolution, each element of the convolution template can be regarded as a weighted Gaussian kernel function. During the network training, their parameters $\mathbf{w}$ and Θ will be learned from the data. This is different from the conventional CNN where each element of the convolution template is a scalar value, instead of a kernel function. We detail this in Fig. 3.

2.4 Network Structure and Postprocessing

Our parcellation network structure and its parameters are presented in Fig. 4. It includes 3 convolution layers and a soft-max layer for the parcellation labels prediction. Since the convolution will gradually increase the reception field, the input patch size needs to be large enough for subsequent convolution operations in deep layers. Herein, the convolution size is chosen as 3 mm, and thus a 10 mm geodesic patch would be large enough to support 3 convolution layers. It is worth noting that the latent Gaussian parametrization kernels are consistently imposed over all convolution layers, which is for preserving spatial metric over intrinsic patches. In addition, since each vertex is independently parcellated, spatially inconsistent labels could be produced. To address this issue, we have adopted graph cuts method to explicitly impose the spatial consistency [6] in parcellation results.

3 Experiments

To validate our method, we manually labeled 90 neonatal cortical surfaces following the FreeSurfer Deskian parcellation protocol. The entire dataset was partitioned into 3 groups for a 3-fold validation. We used two groups for training and the remaining group for testing. We extracted patches from cortical surfaces and empirically set the maximal geodesic distances as 10 mm, which covers a large enough region for the increased reception field. To address inter-subject variation in cortical size, for different subjects, we normalized this distance according to the ratio between the individual brain size and the atlas brain size [13]. The curvature threshold is set as 0.2, and any vertices with absolute mean curvature larger than the threshold will be sampled in a dense manner (with 90% being sampled). Otherwise, it is sampled in a ratio of 0.5. The neural network is implemented in TensorFlow with Adadelta as the optimizer and trained for 40 epochs with a batch size of 100.

Table 1 shows the Dice score of our parcellation methods, compared to other parcellation methods. It can be seen that our method achieves the state-of-the-art performance. Of note, the advantage of our method is that it does not need any spherical mapping and registration. Although the methods in Wu et al. [6] and Zhao et al. [7] also do not rely on spherical registration, they still need to map cortical surfaces onto a sphere. They are not applicable on impaired cortical surfaces with non-spherical topology, because spherical mapping is sensitive to topological noises and cortical surfaces need to be topologically correct before mapping. Generally, the spherical mapping and topology correction of the cortical surface would cost several hours and many researchers are trying to reduce the mapping time [14]. In contrast, our method does not need them and takes about totally an hour including the geodesic distance computation.

To validate whether our method can also handle the impaired cortical surfaces (such as cortical surface after surgery of tumor removal) violating the spherical topology, we

generated a synthetic dataset from normal brains. For 90 subjects in our dataset, we manually removed some parts of brain tissues in the segmentation images, and then reconstruct the impaired white-matter cortical surfaces as shown in Fig. 5(d) and (e). For the impaired cortical surfaces, we can also obtain their corresponding cortical attributes and extract intrinsic patches. Then, we applied the previously-trained models to parcellate these impaired cortical surfaces. It is worth noting that the spherical mapping is not feasible for those impaired cortical surfaces, since they are topologically incorrect or the spherical mapping will produce strange cortical attribute patterns on the mapped sphere. We present our parcellation results on the impaired cortical surface in Fig. 5(d) and (e). It can be seen that our method still achieves good parcellation results on the impaired cortical surface. Since the impaired cortical surface sits in the same space with the intact cortical surface, we can compare the parcellation labels between the impaired cortical surface and the intact cortical surface in a nearest neighbor principle, in order to have a quantitative measurement. Specifically, for any vertex on the impaired cortical surface, we found its nearest vertex on the intact cortical surface (If the distance between these two vertices is smaller than 0.5 mm, they are regarded as the corresponding vertices). Based on the corresponding vertices, we can further compute the overlap ratio between the parcellation labels predicted on the impaired cortical surface and on the intact cortical surface. For our synthetic dataset, the average overlap ratio achieves 0.94 ± 0.03 , indicating almost the same parcellation labels as the intact surface.

4 Conclusion

In this paper, we have proposed an automatic cortical surface parcellation method to work directly on the original cortical surface manifold. We firstly extract the informative intrinsic patches on the cortical surface, and then train a graph convolutional neural network (GCNN) to learn the nonlinear mapping from the cortical attribute patterns (defined on extracted patches) to their corresponding parcellation labels. Experimental results on the normal cortical surfaces show that our method achieves the state-of-the-art parcellation performance. More importantly, our method can also work on impaired cortical surfaces with incorrect topology, where the traditional spherical surface based parcellation methods fail.

Acknowledgements.

This work was partially supported by NIH grants (MH107815, MH116225, and MH117943).

References

1. Glasser MF, et al.: A multi-modal parcellation of human cerebral cortex. *Nature* 536, 171–178 (2016) [PubMed: 27437579]
2. Li G, et al.: Computational neuroanatomy of baby brains: a review. *Neuroimage* 185, 906–925 (2019) [PubMed: 29574033]
3. Desikan RS, et al.: An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *Neuroimage* 31, 968–980 (2006) [PubMed: 16530430]
4. Yang F, Kruggel F: Automatic segmentation of human brain sulci. *Med. Image Anal* 12, 442–451 (2008) [PubMed: 18325826]

5. Li G, Shen D: Consistent sulcal parcellation of longitudinal cortical surfaces. *NeuroImage* 57, 76–88 (2011) [PubMed: 21473919]
6. Wu Z, et al.: Registration-free infant cortical surface parcellation using deep convolutional neural networks In: Frangi AF, Schnabel JA, Davatzikos C, Alberola-López C, Fichtinger G (eds.) *MICCAI 2018. LNCS*, vol. 11072, pp. 672–680. Springer, Cham (2018). 10.1007/978-3-030-00931-177
7. Zhao F, et al.: Spherical U-Net on cortical surfaces: methods and applications In: Chung ACS, Gee JC, Yushkevich PA, Bao S (eds.) *IPMI 2019. LNCS*, vol. 11492, pp. 855–866. Springer, Cham (2019). 10.1007/978-3-030-20351-167
8. Gopinath K, et al.: Graph convolutions on spectral embeddings for cortical surface parcellation. *Med. Image Anal* 54, 297–305 (2019) [PubMed: 30974398]
9. Cucurull G, et al.: Convolutional neural networks for mesh-based parcellation of the cerebral cortex In: *Proceedings of MIDL* (2018)
10. Fischl B, et al.: High-resolution intersubject averaging and a coordinate system for the cortical surface. *Hum. Brain Mapp* 8, 272–284 (1999) [PubMed: 10619420]
11. Mitchell JS, et al.: The discrete geodesic problem. *SIAM J. Comput* 16, 647–668 (1987)
12. Monti F, et al.: Geometric deep learning on graphs and manifolds using mixture model CNNs In: *Proceedings of CVPR*, pp. 5115–5124 (2017)
13. Wu Z, et al.: Construction of spatiotemporal neonatal cortical surface atlases using a large-scale dataset In: *Proceedings of ISBI*, pp. 1056–1059. IEEE (2018)
14. Choi PT, et al.: Flash: fast landmark aligned spherical harmonic parameterization for genus-0 closed brain surfaces. *SIAM J. Imaging Sci* 8, 67–94 (2015)

Fig. 1.

The cortical surface manifold with color-coded cortical attributes maps (a-d) and parcellation labels map (e).

Fig. 2.

The informative local intrinsic patch extraction. (a) The patch sampling positions on the cortical surface. These positions are determined by the mean curvature map. The white and black vertices indicate the locations with high and low absolute curvatures, respectively. (b) The minimal principal curvature direction field of the enclosed local region in (a). (c) The local intrinsic patch and its charting strategy. Red circle is the iso-geodesic-distance contour of v_i bounded by ρ_{max} . Red arrow is the minimal curvature direction. ρ and θ are charting coordinates.

Fig. 3.

Illustration of the convolution on the graph. The blue triangles are the typical vertex v_k^i included in patch P_{v_i} . $\mathcal{N}_j, j = 1, \dots, 9$, is the Gaussian kernels for parameterizing this patch. w_j is the corresponding convolution weight. Therefore, each element of the convolution template can be regarded as a weighted kernel function, i.e., $w_j \mathcal{N}_j$, which will be learned during the training.

Fig. 4.
Parcellation network structure and corresponding parameters.

Fig. 5.

Parcellations on a typical subject. (a) The manually labeled parcellation. (b) Parcellation using our proposed GCNN on a normal brain. (c) Parcellation using our proposed GCNN on a normal brain with the graph-cut to impose spatial consistency. (d) Parcellation using our proposed GCNN on an impaired cortical surface, where the red box indicates the impaired region and has been zoomed up for better inspection. (e) Parcellation results using our proposed GCNN on an impaired cortical surface with also the graph-cut to impose the spatial consistency.

Table 1.

The overall parcellation performance of different parcellation methods.

Method	Dice score
Spherical registration + Majority voting	84.54 ± 0.08
Spherical CNN + graph cut [6]	87.06 ± 0.06
Spherical UNet [7]	88.69 ± 0.06
GCNN without graph cut (ours)	86.48 ± 0.06
GCNN with graph cut (ours)	87.88 ± 0.06

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript