SegRecon: Learning Joint Brain Surface Reconstruction and Segmentation from Images

Karthik Gopinath, Christian Desrosiers, and Herve Lombaert

ETS Montreal, Canada

Abstract. Commonly-used tools for cortical reconstruction and parcellation, such as FreeSurfer, are central to brain surface analysis but require extensive computation times. This paper proposes SegRecon, a fast learning approach where an integrated end-to-end deep learning method does *simultaneously* reconstruct and segment cortical surfaces directly from an MRI volume, all in a single step. We train a volume-based neural network to predict, for each voxel, the signed distance to the white-to-grey-matter interface along with its corresponding spherical representation in the registered atlas space. The continuous representation of the spherical coordinates enables our approach to naturally extract an implicit isolevel surface for its reconstruction and obtain the parcel labels from the spherical atlas. We illustrate the advantages of our method with thorough experiments on the MindBoggle dataset. Our parcellation results show more than 4% improvements in average Dice accuracy with respect to FreeSurfer and a drastic speed-up from hours to seconds of computation.

Keywords: Brain surface reconstruction · Surface segmentation · Parcellation.

1 Introduction

The accurate reconstruction and segmentation of cortical surfaces from MRI are essential to a variety of brain analyses [23,8]. Standard pipelines for surface reconstruction [6,4,13,17,24] follow a sequence of costly operations that often include: white matter segmentation, surface mesh generation from the segmentation masks, mesh smoothing and projection to a sphere, topological correction of the projected mesh, and fine-tuning of re-projected mesh on the segmented volume. The segmentation of the cortical surface into neuroanatomical parcels is then performed in a subsequent and typically more expensive step of up-to 4 hours, which involves the re-projection of each surface to a sphere via a metric-preserving inflation process, registration to a spherical atlas [7,16] and cortical parcellation using atlas labels [5].

Recently, Henschel et al. [12] developed a pipeline called FastSurfer that accelerates processing times using deep learning for brain segmentation and spectral embedding for registration to a spherical atlas. Despite considerably reducing processing times compared to traditional approaches, this pipeline still requires the processing of volume segmentation and surface reconstruction in two consecutive steps, via a combination of different techniques. To overcome this limitation, Cruz et al. [3] has proposed a deep learning model for cortical surface reconstruction, called DeepCSR. Inspired by [21], DeepCSR reconstructs a surface without the need for an explicit segmentation, by sampling points on a reference grid of arbitrary resolution. However, this process is highly expensive for surfaces with hundreds of thousands of points, in terms of both computation and memory. Additionally, DeepCSR *only* enables surface reconstruction, and *not* parcellation which is one of the most time-costly operations in standard neuroimaging pipelines. Approaches in the lines of [9,25,18,10,19] have proposed algorithms to operate directly on surface data for cortical parcellation. Spectral embeddings of surface meshes in a low-dimensional space [18] were also used to predict cortical parcellation labels, but their main limitation was that mesh nodes are considered separately instead of jointly. To better exploit the connectivity information of a mesh graph, recent work has been proposed with graph convolutional networks (GCN) [9,25,11,10]. While this strategy provides a faster and more accurate parcellation of the cortical surface, it is, however, sensitive to any error from a separate surface mesh reconstruction.

This paper proposes a novel deep learning model, SegRecon, for the joint reconstruction and parcellation of cortical surfaces. Our end-to-end model works directly on MRI volumes and predicts a dense set of surface points along with their corresponding parcellation labels. The proposed architecture, built upon the 3D-UNet [2], is trained to predict for each voxel of an input volume a vector encoding the brain hemisphere of the voxel, its signed distance to the white-to-grey-matter interface, and its spherical coordinates in the registered atlas space. By learning this multi-task problem, the network can thereafter be used to reconstruct and segment implicit brain surfaces both efficiently and topologically accurate [1].

The main contributions of our work are the following: **(1)** To our knowledge, we propose the first deep learning model to take an MRI volume as input and parcelate and jointly reconstruct the brain surfaces. This contrasts with existing approaches, which either perform surface reconstruction and segmentation in separate steps [12], are limited to reconstruction [3], or require a pre-generated mesh as input [9,25,18]; **(2)** Compared to DeepCSR, the proposed network implements a fully-convolutional architecture that densely predicts the location relative to cortical surfaces for each voxel of the input image, in a single feed-forward pass; **(3)** For parcellation, with respect to the widely-used FreeSurfer software, our method achieves a 4.3% improvement in Dice, while being several orders of magnitude faster as well as generating cortical surface.

In the next section, we present the proposed deep learning model for these tasks, describing in detail the network architecture, training losses, and inference steps. The performance of our method is evaluated on the MindBoggle dataset [15]. The ablation study and comparison to the state-of-art in our experiments demonstrate the benefits of our method.

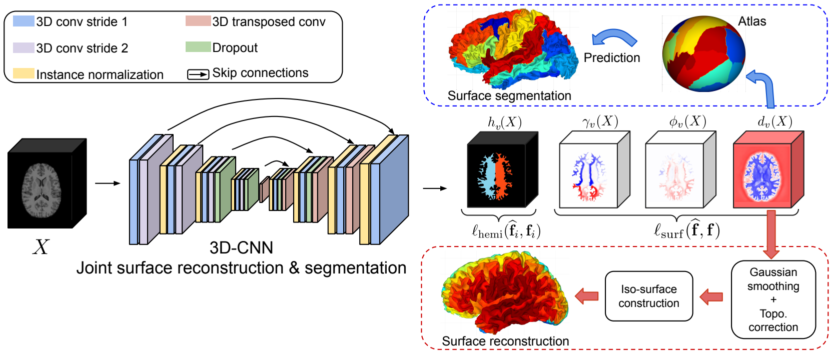

Fig. 1. Overview of SegRecon: The 3D-CNN model takes as input MRI volume X for joint learning of the signed distance to white-to-grey matter interface and its corresponding spherical coordinates in the registered atlas space. (*Red*) The cortical surface is reconstructed by applying Gaussian smoothing and topological correction on the predicted signed distance map prediction $d_v(X)$, followed by iso-surface reconstruction via the Marching Cubes algorithm. (*Blue*) In parallel, the predicted spherical atlas coordinates ($\gamma_v(X)$, $\phi_v(X)$) and hemisphere label ($h_v(X)$) are used to propagate atlas parcellation labels to near-surface voxels v. An illustration for left hemisphere is shown here.

2 Method

An overview of SegRecon is shown in Fig. 1 with the end-to-end surface construction and segmentation steps illustrated. Let $\mathcal{D} = \{(X_i, S_i, Y_i)\}_{i=1}^{n}$ be a training set composed of 3D volumes, $X \in \mathbb{R}^{|\Omega|}$ with voxel set $\Omega \subset \mathbb{R}^3$; surfaces defined by m points, $S_i \in \mathbb{R}^{m\times 3}$; and segmentation labels, $Y_i \in \mathbb{R}^{m\times c}$ where c is the number of segmentation classes. The goal is to learn a function f parameterized by θ which maps an input 3D volume X to a surface with points S and corresponding labels Y.

One of the main challenges in this task comes from the disparity between the well-defined grid space of images X and the domain of surfaces S where the number of points can vary from one surface to another and points can lie anywhere in 3D space. In [3], this problem is solved by giving as input to model f both the image X and a query point $p \in \mathbb{R}^3$ in the template space. The model then predicts if p belongs to the surface in X, or alternatively its distance to this surface. To reconstruct a surface at inference time, the model is queried over a fixed reference grid. While this strategy allows reconstructing a surface at arbitrary resolution, it suffers from two important drawbacks. First, since the template points which can be in the hundreds of thousands are queried independently, reconstructing a surface requires significant time and computation. Moreover, unlike dense prediction approaches, this strategy does not exploit the spatial relationship between points. Last, because feature maps need to be computed for the whole 3D volume X, it also needs a large amount of memory.

To overcome these drawbacks, we instead learn a model that densely projects voxels of the input volume X to a spherical atlas space. Specifically, f maps each voxel $v \in \Omega$ to a vector

$$f_v(X) \;=\; \big[d_v(X), \phi_v(X), \gamma_v(X), h_v^{lh}(X), h_v^{rh}(X), h_v^{bg}(X)\big], \tag{1}$$

where $d_v(X)$ is the signed distance from v to its nearest surface point, such that $d_v(X) \leq 0$ if v is inside the surface else $d_v(X) > 0$, $\phi_v(X)$, $\gamma_v(X)$ are the polar angle and azimuthal angle of $v \in \Omega$ defining its position in the spherical atlas, and $h_v^{lh}(X)$, $h_v^{rh}(X)$, $h_v^{bg}(X) \in [0, 1]$ are the probabilities that v is in the left hemisphere, right hemisphere and background, respectively. Here, polar and azimuthal angles are normalized so to lie in the $[-1, 1]$ range. A further topological correction step [1] over the predicted surface points prevents the extraction of critical points yielding topological defects. The resulting surface is defined implicitly as the 0-levelset of the distance map and can be efficiently reconstructed using an iso-surface extraction algorithm such as the Marching Cubes [20].

2.1 Training the model

Denote $\widehat{\mathbf{f}}_i = f(X_i)$ as the predicted vector for an image X_i and let $\mathbf{f}_i$ be the corresponding ground-truth. To train the model, we use the following loss function

$$\mathcal{L}(\theta; \mathcal{D}) \;=\; \sum_{i=1}^{n} \ell_{\mathrm{surf}}(\widehat{\mathbf{f}}_i, \mathbf{f}_i) \;+\; \lambda\, \ell_{\mathrm{hemi}}(\widehat{\mathbf{f}}_i, \mathbf{f}_i), \tag{2}$$

The first loss term, ℓ_{surf}, ensures that the signed distance of voxels to the surface, as well as their position in the spherical atlas space, are well predicted. Dropping index i for simplicity, it is defined as:

$$\begin{aligned}\ell_{\mathrm{surf}}(\widehat{\mathbf{f}}, \mathbf{f}) \;=\; \sum_{v \in \Omega} \mathbb{1}_{|d_v| \leq \epsilon} \cdot \Big[(\widehat{d}_v - d_v)^2 \;+\; \min\big\{(\widehat{\phi}_v - \phi_v)^2,\, (1 + \widehat{\phi}_v - \phi_v)^2\big\} \\ +\; \min\big\{(\widehat{\gamma}_v - \gamma_v)^2,\, (1 + \widehat{\gamma}_v - \gamma_v)^2)\big\}\Big].\end{aligned} \tag{3}$$

where $\mathbb{1}_P$ is the indicator function, equal to 1 if predicate P is true else, 0 otherwise. We only consider voxels within a distance of ϵ to the nearest surface point in order to focus learning on relevant points close to our surface. This is achieved with function $\mathbb{1}_{|d_v| \leq \epsilon}$ in Eq. (3). Additionally, we consider the non-uniqueness of spherical coordinates (e.g., $-\pi \equiv \pi$) by computing, for each angle, the minimum L_2 distance from the predicted angle or this angle plus 1 to the ground-truth. The distance d_v is, therefore, defined between the center of the voxel v in image space and the nearest point on surface S. In this work, we use the surface mesh generated by FreeSurfer for training. The sign of d_v is determined using the white-matter segmentation mask, with voxels inside the white matter having a negative distance. Likewise, the ground-truth spherical coordinates ϕ_v and γ_v are obtained using FreeSurfer [6] with the Desikan-Killiany-Tourville (DKT) atlas [16].

The second term, ℓ_{hemi} enables the network to predict in which hemisphere lies a voxel v. This prediction is necessary since the surface atlas is defined separately for each hemisphere. Here, we use cross-entropy as loss function:

$$\ell_{\text{hemi}}(\widehat{\mathbf{f}}, \mathbf{f}) \;=\; -\sum_{v \in \Omega} \sum_{c \in \{lh, rh, bg\}} h_v^c \log \widehat{h}_v^c. \tag{4}$$

The ground-truth hemisphere masks are once again obtained from FreeSurfer.

2.2 Surface reconstruction and segmentation

Once the network is trained, it can be used to reconstruct and segment surfaces directly from a test volume X. First, we feed the volume to the network to obtain a prediction vector for all voxels. Since the network is fully-convolutional, this can be done efficiently in a single feed-forward pass. Next, we apply a small-width Gaussian filter on the predicted 3D distance map $\widehat{d}$ using a single convolution operation and employ a topological correction step [1] to overcome any defects in the surface.

To segment the surface, we first compute the near-surface voxels in each hemisphere as follows:

$$S^c \;=\; \big\{ v \in \Omega \mid |d_v| \leq \epsilon \,\wedge\, c = \arg\max_{c'} \widehat{h}_v^{c'} \big\}, \quad c \in \{lh, rh\}. \tag{5}$$

We then find the nearest-neighbor to a given reference atlas R^c for all the near-surface voxels $v \in S^c$ using their predicted angles $\widehat{\phi}_v$ and $\widehat{\gamma}_v$. The segmentation labels from this reference atlas R^c are then projected back to the near-surface voxels S^c.

2.3 Implementation details

The overall architecture of SegRecon is shown in Fig.1. As an input, we provide the skull-stripped, intensity normalized 3D T1-MRI volume. We use a 3D-UNet architecture similar to [2] in order to map the input voxel to a point in the spherical atlas space.

We apply a softmax activation in the first three output channels to predict the probability of a voxel belonging to the background, left hemisphere, or right hemisphere. The polar and azimuthal angles, $\widehat{\phi}_v$ and $\widehat{\gamma}_v$, are predicted with a tanh activation. The last output channel produces the signed distance map $\widehat{d_v}$ for each voxel. The network parameters, θ, are optimized using a stochastic gradient descent with the Adam optimizer[14]. During training, we pick ϵ in Eq. 3 to be 2.5. The surface is reconstructed using the Marching Cubes [20] on the 0-levelset of a distance map of the predicted signed distance, smoothed with a Gaussian kernel of sigma=0.5 and topologically correct [1]. We use an i7 desktop machine with 16Gb RAM and Nvidia RTX 2080 GPU for our work.

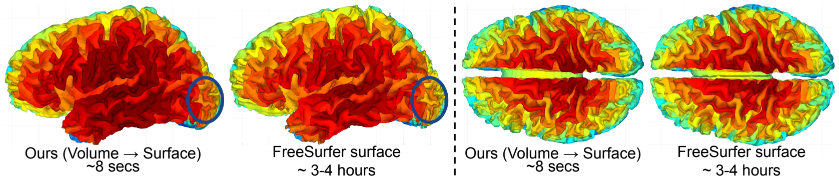

Fig. 2. Surface visualization: Comparison of a cortical surface predicted by our method (Gaussian smoothing $\sigma = 0.001$) and FreeSurfer [7]. Our SegRecon method in comparison with FreeSurfer meshes yields a reconstruction error of 1.313 voxels (avg. Hausdorff distance) with visually similar results while being orders of magnitude faster.

3 Experiments and results

To benchmark the performance of our method, we use one of the largest publicly available dataset containing manual surface parcellation, MindBoggle [15]. This dataset contains 101 subjects with MRI volumes, FreeSurfer processed meshes, and 32 manually-labeled cortical parcels. We split the dataset randomly into training, validation, and testing using a ratio of 70-10-20%. The qualitative results of the reconstructed surface is shown in 2. The reconstruction error in terms of Hausdorff distances between the predicted and FreeSurfer meshes is found to be 1.313 voxels only. In a first experiment, we evaluate the effect of varying the reference atlas template for predicting parcellation labels, and show that a robust parcellation can be achieved by combining the predictions from multiple atlases. Finally, we highlight the advantages of joint reconstruction and parcellation methods against state-of-the-art methods.

3.1 Effect of reference atlas on parcellation

Instead of predicting class probabilities for each voxel, as in standard 3D segmentation networks, the proposed network predicts spherical atlas coordinates (i.e., angles $\widehat{\phi}_v$ and $\widehat{\gamma}_v$). This has two important advantages: *i*) considerably reducing the number of outputs for the number of classes to only two, and *ii*) providing information on the precise location of a voxel inside a parcel instead of simply measuring if a voxel is inside a parcel or not. As we will show in the next section, this continuous prediction strategy leads to a higher accuracy compared to a standard segmentation approach. However, the final predicted labels depend on the reference atlas.

For assessing the impact of the reference atlas on segmentation performance, we randomly select five subjects from the training set and use the spherical coordinates and parcellation labels of their surface mesh nodes as different atlases $\mathrm{Ref}_1, \ldots, \mathrm{Ref}_5$. Table 1 reports the mean Dice score obtained for test subjects using each of the five atlases. While a high accuracy is obtained in all cases, the performance also varies significantly from 84.60% to 87.33%.

To provide a greater robustness to the choice of atlas, we apply a simple multi-atlas strategy in which a separate prediction is obtained for each atlas, and individual predictions are then combined using majority voting. As shown in Table 1 (last column) this strategy leads to an important boost in Dice score to 88.69% compared to the average of 85.63% computed across all atlases.

3.2 Comparison with the state-of-the-art

We next compare our joint reconstruction and parcellation method against several baselines and recent approaches. Table 2 reports the performance of tested methods in terms of average Dice scores, mean Hausdorff distances and runtime. To evaluate the benefit of predicting cortical parcels using spherical atlas coordinates, we first train a 3D-UNet to predict the parcellation label probabilities directly at the voxel level as in standard 3D segmentation networks. This baseline, called DirectSeg in Table 2, gives a low Dice score of 79.95%.

We also evaluate the FreeSurfer parcellation against the manual labels provided in the MindBoggle dataset. FreeSurfer considerably improves parcellation accuracy compared to DirectSeg with a Dice score of 84.39%. However, this comes at the price of a significant increase in computation times, from 300 milliseconds per volume for DirectSeg to a few hours for FreeSurfer.

Third, we show the advantage of predicting cortical surfaces directly from 3D images, as in our method, compared to working with surface meshes computed previously. Toward this goal, we test two mesh-based models, named FS + SRF and FS + GCN in the results. The first one, Spectral Random Forest (SRF) [18], performs a spectral embedding of nodes in the FreeSurfer mesh graph using the main eigen-components of its Laplace matrix. The labels of embedded nodes are then predicted separately using a Random Forest classifier. In the latter, the connectivity of nodes in the mesh graph is also exploited in the prediction using a graph convolutional network (GCN) [9]. As can be seen, predicting labels for all nodes simultaneously in FS + GCN, instead of individually in FS + RF, largely improves Dice score by 6.72%. However, as both approaches require generating surface meshes in a former step, which can take around 2 hours for FreeSurfer, their total run time remains substantial. In comparison, our method achieves a mean Dice score of 88.69% with an average total run time of 8 seconds per volume. That is a 4.30% improvement over the Dice score of FreeSurfer, at a fraction of its computational cost.

Next, we evaluate the performance of our method SegRecon in two different settings. First, we show the importance of the hemisphere prediction loss of Eq. (4) on performance. To do so, we reduce the weight $\lambda = 0.0001$ of the

Table 1. Effect of reference atlas: Column 1-5: The average Dice overlap (in %) obtained after using five different references as an atlas for label propagation. The last column shows the results when we vote across five different atlas references.

Ref_1	Ref_2	Ref_3	Ref_4	Ref_5	**Voting**
84.60 ± 1.90	85.85 ± 1.79	85.29 ± 1.93	85.08 ± 1.54	87.33 ± 1.90	**88.69 ± 1.84**

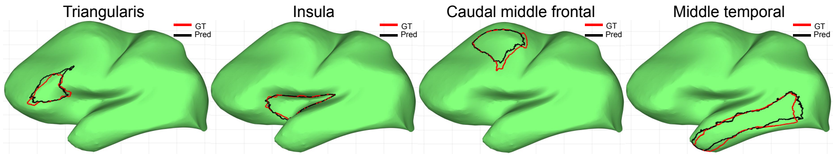

Fig. 3. Parcellation performance: The manual parcellation boundaries are shown in red, with our predicted parcellation boundaries in black. Our model segments 32 parcels in total on the brain surface. We show four parcels, namely, Triangularis, Insula, Caudal middle frontal and middle temporal of the left hemisphere for qualitative analysis. The cortical mesh is inflated here for visualization.

Table 2. Evaluation of SegRecon: Comparison against approaches in terms of Dice scores (in %), Hausdorff Distances (in mm), and computational time. The first row shows the performance of a DirectSeg a 3D-CNN network on surface parcellation. The second row illustrates the results of the traditional FreeSurfer algorithm for parcellation. In the third and fourth row, we show the ability of a Spectral Random Forest (SRF) and GCN learning based approach to segment the cortical surface. The fifth row shows the importance of learning hemisphere segmentation in our work. Finally, in the last row, we show the performance of our model.

Methods	Dice overlap (%)	Hausdorff (mm)	Time
DirectSeg	79.95 ±2.58	–	∼ 300 milliseconds
FreeSurfer	84.39 ±1.91	2.11 ±0.29	∼ 4 hours
FS + SRF	79.89 ±2.62	1.97 ±0.40	∼ 2 hours + 18 sec
FS + GCN	86.61 ±2.45	1.66 ±0.44	∼ 2 hours + 3 sec
w/o hemisphere	59.28 ±12.20	3.94 ±3.14	∼ 8 sec
SegRecon (Ours)	**88.69 ±1.84**	**1.20 ±1.36**	∼ 8 sec

loss term ℓ_{hemi} during training in Eq. (2). This ablation baseline is denoted as *w/o hemisphere* in Table 2. As can be observed, the lack of accurate hemisphere prediction results in ambiguous label prediction for surface voxels in both hemispheres which results in a low Dice score of 59.28%. Finally, we present the setting of our model for predicting a distance $\widehat{d_v}$ for each voxel. In this way, our model predicts the iso-surface for surface reconstruction. The accurate prediction of polar and azimuthal angles (i.e., $\widehat{\phi_v}$ and $\widehat{\gamma_v}$) for obtaining parcel labels from the atlas yields an average Dice score of 88.69%. Similar improvements of our method compared to other approaches are also found for the Hausdorff distance metric. Qualitative results obtained by our surface segmentation method are shown in Fig. 3, where we illustrate the differences between the predicted and manual label boundaries for four different parcels or regions.

4 Conclusion

In this work, we presented SegRecon, a novel deep learning end-to-end model for the joint surface reconstruction and segmentation directly from MRI volumes. The signed distance map predicted densely by our network offers a implicit description of the surface. After enforcing a topological guarantee [1], a surface mesh is generated from the signed distance map using, for instance, the Marching Cubes [20]. Our experiments first analyzed the impact of the reference atlas used for transferring cortical parcellation labels to the surface. A robust performance with Dice score of 88.69% can be achieved for cortical parcellation via a multi-atlas strategy where the predictions for different atlases are combined using majority voting. We also compared our method against several baselines and state-of-the-art approaches for cortical parcellation. Our approach has higher Dice score over 3D-UNet (79.9%), without offering a reconstructed cortical surface, higher Dice score (84.3%) and lower computation time over FreeSurfer (hours vs. seconds) and can perform a joint surface reconstruction and parcellation with higher Dice score compared to GCN (86.6%) that require pre-computed surfaces as input. Furthermore, while state-of-art methods obtained a mean Hausdorff distance close to 2 mm, our model achieved a lowest mean distance of 1.20 mm. While Gaussian smoothing and topological correction can alleviate reconstruction artifacts, this technique are performed as a post processing step. Additionally, imposing topological constraints during network training and local smoothing based on anisotropic diffusion [22] could help regularize the mesh while better preserving cortical folding patterns.

References

1. Bazin, P.L., Pham, D.L.: Topology correction of segmented medical images using a fast marching algorithm. Computer methods and programs in biomedicine (2007)
2. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: MICCAI (2016)
3. Cruz, R.S., Lebrat, L., Bourgeat, P., Fookes, C., Fripp, J., Salvado, O.: Deepcsr: A 3d deep learning approach for cortical surface reconstruction. arXiv preprint arXiv:2010.11423 (2020)
4. Dahnke, R., Yotter, R.A., Gaser, C.: Cortical thickness and central surface estimation. Neuroimage (2013)
5. Desikan, R.S., Ségonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., et al.: An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. Neuroimage (2006)
6. Fischl, B., et al.: Automatically parcellating the cortex. Cereb. Cortex (2004)
7. Fischl, B., Sereno, M.I., Dale, A.M.: Cortical surface-based analysis: Ii: inflation, flattening, and a surface-based coordinate system. Neuroimage (1999)
8. Glasser, M.F., Coalson, T.S., Robinson, E.C., Hacker, C.D., Harwell, J., Yacoub, E., Ugurbil, K., Andersson, J., Beckmann, C.F., Jenkinson, M., et al.: A multi-modal parcellation of human cerebral cortex. Nature (2016)
9. Gopinath, K., Desrosiers, C., Lombaert, H.: Graph convolutions on spectral embeddings for cortical surface parcellation. Medical image analysis (2019)
10. Gopinath, K., Desrosiers, C., Lombaert, H.: Graph domain adaptation for alignment-invariant brain surface segmentation. arXiv preprint arXiv:2004.00074 (2020)

11. He, R., Gopinath, K., Desrosiers, C., Lombaert, H.: Spectral graph transformer networks for brain surface parcellation. In: ISBI (2020)
12. Henschel, L., Conjeti, S., Estrada, S., Diers, K., Fischl, B., Reuter, M.: Fastsurfer-a fast and accurate deep learning based neuroimaging pipeline. NeuroImage (2020)
13. Kim, J.S., Singh, V., Lee, J.K., Lerch, J., Ad-Dab'bagh, Y., MacDonald, D., Lee, J.M., Kim, S.I., Evans, A.C.: Automated 3-d extraction and evaluation of the inner and outer cortical surfaces using a laplacian map and partial volume effect classification. Neuroimage (2005)
14. Kingma, D.P., Ba, J.: Adam: Stochastic optimization. In: ICLR (2014)
15. Klein, A., Ghosh, S.S., Bao, F.S., Giard, J., Häme, Y., Stavsky, E., Lee, N., Rossa, B., Reuter, M., Chaibub Neto, E., Keshavan, A.: Mindboggling morphometry of human brains. PLOS Computational Biology (2017)
16. Klein, A., Tourville, J.: 101 labeled brain images and a consistent human cortical labeling protocol. Frontiers in neuroscience (2012)
17. Kriegeskorte, N., Goebel, R.: An efficient algorithm for topologically correct segmentation of the cortical sheet in anatomical mr volumes. NeuroImage (2001)
18. Lombaert, H., Criminisi, A., Ayache, N.: Spectral forests: Learning of surface data, application to cortical parcellation. In: MICCAI (2015)
19. López-López, N., Vázquez, A., Poupon, C., Mangin, J.F., Ladra, S., Guevara, P.: Geosp: A parallel method for a cortical surface parcellation based on geodesic distance. In: EMBC (2020)
20. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. ACM siggraph computer graphics (1987)
21. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: CVPR (2019)
22. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on pattern analysis and machine intelligence **12**(7), 629–639 (1990)
23. Querbes, O., Aubry, F., Pariente, J., Lotterie, J.A., Démonet, J.F., Duret, V., Puel, M., Berry, I., Fort, J.C., Celsis, P., et al.: Early diagnosis of alzheimer's disease using cortical thickness: impact of cognitive reserve. Brain (2009)
24. Shattuck, D.W., Leahy, R.M.: Brainsuite: an automated cortical surface identification tool. Medical image analysis (2002)
25. Wu, Z., Zhao, F., Xia, J., Wang, L., Lin, W., Gilmore, J.H., Li, G., Shen, D.: Intrinsic patch-based cortical anatomical parcellation using graph convolutional neural network on surface manifold. In: MICCAI (2019)