Efficient Cortical Surface Parcellation via Full-Band Diffusion Learning at Individual Space

Yuanzhuo Zhu[1], Chunfeng Lian[2,4(✉)], Xianjun Li[3], Fan Wang[1,3(✉)], and Jianhua Ma[1,4(✉)]

[1] Key Laboratory of Biomedical Information Engineering of Ministry of Education, School of Life Science and Technology, Xi'an Jiaotong University, Xi'an, China
fan.wang@xjtu.edu.cn

[2] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, China
chunfeng.lian@xjtu.edu.cn

[3] Department of Radiology, The First Affiliated Hospital of Xi'an Jiaotong University, Xi'an, China

[4] Pazhou Lab (Huangpu), Guangzhou, China
jhma@xjtu.edu.cn

Abstract. Cortical parcellation delineates the cerebral cortex into distinct regions based on anatomical and/or functional criteria, a process crucial for neuroscientific research and clinical applications. Conventional methods for cortical parcellation involve spherical mapping and complex feature computation, which are time-consuming and prone to error. Recent geometric learning approaches offer some improvements but may still depend on spherical mapping and could be sensitive to mesh variations. In this work, we present Cortex-Diffusion, a fully automatic framework for cortical parcellation on native cortical surfaces without spherical mapping or morphological feature extraction. Leveraging the DiffusionNet as its backbone, Cortex-Diffusion integrates a newly designed module for full-band spectral-accelerated spatial diffusion learning to adaptively aggregate information across highly convoluted meshes, allowing high-resolution geometric representation and accurate vertex-wise delineation. Using only raw 3D vertex coordinates, the model is compact, with merely 0.49 MB of learnable parameters. Extensive experiments on adult and infant datasets demonstrates that Cortex-Diffusion achieves superior accuracy and robustness in cortical parcellation. Our code is available at https://github.com/ladderlab-xjtu/CortexDiffusion.

Keywords: Cortical surface parcellation · Geometric deep learning · Learnable spatial diffusion · Full-band spectral acceleration.

1 Introduction

Cortical parcellation of the cerebral cortex entails segregating the cortical surface into distinct regions or parcels, based on information such as anatomical/functional characteristics. This segmentation process is critical across various fields, especially neuroscience, neurology, and psychology. Reliable and accurate parcellation serves as a fundamental prerequisite for exploring the anatomical

and functional organization of the brain, facilitating surgical planning, and enhancing the understanding of the effects of neurological disorders.

In practice, multiple computational tools are publicly available for neuroimage-based cerebral cortex analyses [3,6,14,20,17]. After the reconstruction of the mesh surface for individual hemispheres, standard software packages usually execute parcellation in three steps. Initially, a hemisphere characterized by its intricate and highly convoluted form is expanded and projected onto a spherical shape. Subsequently, this sphere undergoes a non-linear alignment with an atlas, informed by various morphological characteristics. The final parcellation derives from adjusting the atlas's pre-defined parcellation maps to the original surface, utilizing the transformation field generated in the prior step. Despite demonstrating commendable outcomes, two principal challenges exist: 1) The involvement of multiple mapping and registration phases may lead to the introduction of unnecessary errors; 2) The process of spherical mapping and registration, along with the computation of cortical features, is notably time-intensive, typically requiring about 10 minutes per hemisphere on the latest PC.

Some geometric learning-based methods have been proposed to improve the efficiency in parcellation, e.g., Spherical U-Net [19] and SPHARM-Net [7]. These methods learn feature embeddings on the spherical surface, eliminating the necessity for registration. Besides, techniques from the computer graphics community, e.g., SubdivNet [9] and MeshCNN [8], demonstrate the capability to effectively learn on native surfaces without the need for converting formats, suggesting their potential applicability to cortical surface parcellation tasks. Nevertheless, these approaches often still require the explicit calculation of morphological features and/or the use of spherical mapping. Meanwhile, most of them tend to be susceptible to variations in mesh types and resolutions, experiencing notable performance degradations across varying situations.

In this paper, we introduce a novel framework for the task of cortical parcellation directly on native surfaces, eliminating the need for spherical mapping and morphological feature quantification. Our method employs DiffusionNet [16] as its backbone, incorporating a newly designed module capable of learning full-band spectral-accelerated spatial diffusion for adaptive information aggregation on intricate and highly convoluted meshes. This module is adept at capturing high-resolution, detailed geometric information necessary for accurately distinguishing between various cortical regions on a vertex-by-vertex basis. Termed **Cortex-Diffusion**, our approach relies solely on the raw 3D coordinates of vertices as input, resulting in an efficient model that requires only 0.49 MB of learnable parameters. Experiments on both adult and infant datasets show that Cortex-Diffusion consistently achieved leading performance in cortical parcellation, demonstrating exceptional accuracy and stability.

2 Cortex-Diffusion

2.1 Architecture

In this work, we introduce an advanced learning-based method, referred to as Cortex-Diffusion, designed for the precise parcellation of high-resolution cor-

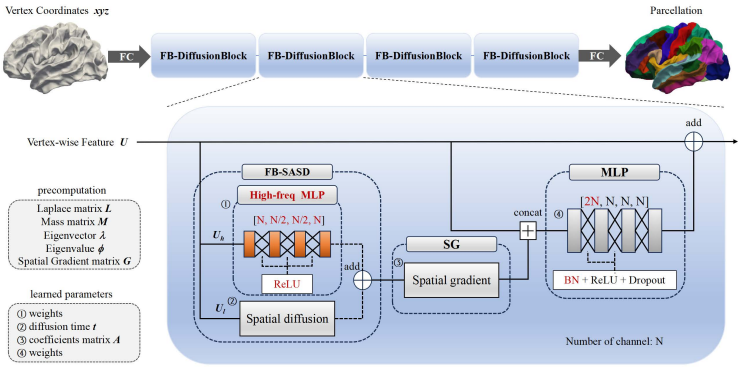

Fig. 1. The schematic diagram of our Cortex-Diffusion for cortical surface parcellation.

tical surface in individual subjects (i.e., within their native space). Cortex-Diffusion integrates the pioneering DiffusionNet [16] as its backbone to establish a lightweight geometric deep network, characterized by a minimalistic design that requires only a small number of learnable parameters. It processes the raw vertical coordinates of a meshed cortical surface, utilizing these inputs to hierarchically extract fine-grained geometric representations, enabling the accurate assignment of each vertex to its respective parcel. A notable advantage of our method is its high efficiency, obviating the need for the time-intensive processes of spherical mapping, registration, and attribute quantification. As the schematic diagram shown in Fig. 1, the architecture of Cortex-Diffusion is compact, which only contains four cascaded FB-Diffusion blocks. In each block, there are three fundamental modules, including **1**) a *vertex-wise multi-layer perceptron* (MLP) shared across vertices for nonlinear feature mapping, **2**) a *full-band spectral-accelerated spatial diffusion* (FB-SASD) module for channel-wise feature propagation across adaptively identified geodesic neighborhood, and **3**) a *spatial gradient* (SG) operation that learns anisotropic feature extraction.

Specifically, the MLP module in each FB-Diffusion block consists of three vertex-wise linear layers, where each of the first two is followed by batch normalization (BN), rectified linear unit (ReLU) activation, and dropout.

The SG module is implemented the same as that in the original DiffusionNet. Let S be a specific mesh with totally V vertices, and its input feature matrix for a particular SG module is $\mathbf{U} \in \mathbb{R}^{V\times N}$, where N denotes the number of feature channels. The SG module first quantifies the spatial gradients of $\mathbf{U}$ per channel according to the specific topology of S, based on which an $N \times N$ matrix $\boldsymbol{A}$ is learned to scale and rotate $\boldsymbol{U}$ across channels for directional feature enhancement along the surface of S. More specifically, the channel-wise feature gradients (say $\mathbf{Z} \in \mathbb{C}^{V\times N}$) are determined by $\mathbf{Z} = \mathbf{GU}$, where $\mathbf{G} \in \mathbb{C}^{V\times V}$ is a mesh-specific spatial gradient matrix that stores the 2D gradient vectors (i.e., complex numbers here to facilitate following calculations) quantified in the tangent plane for each vertex of S. Based on $\mathbf{Z}$, SG finally updated the input feature tensor as

$$\mathbf{U}' = tanh(Re(\overline{\mathbf{Z}} \odot \mathbf{ZA})), \tag{1}$$

where $\overline{\mathbf{Z}}$ is the conjugate of $\mathbf{Z}$, $\odot$ denotes the Hadamard product, $Re(\cdot)$ returns the real part of a complex tensor, and $tanh(\cdot)$ is the element-wise tanh activation. Notably, such an SG module *only has one learnable part*, i.e., matrix $\mathbf{A}$.

The most important technical contribution of our Cortex-Diffusion compared with the original DiffusionNet is on the design of the FB-SASD module for more adaptive local information aggregation on cortical surfaces with highly complex shapes. We present these details in the subsequent subsection (i.e, **Sec.** 2.2).

2.2 Learnable Full-Band Spectral-Accelerated Spatial Diffusion

In the discretized setting of triangle meshes, diffusing information over a surface S can be modeled by the discretization of a continuous heat function. Specifically, given the input $\mathbf{U} \in \mathbb{R}^{V \times N}$, the diffusion of each feature channel can be

$$H_{t_n}(\mathbf{U}[:,n]) = exp(-t_n \mathbf{M}^{-1}\mathbf{L})\mathbf{U}[:,n], \forall n = 1, \ldots, N, \tag{2}$$

where $\mathbf{L} \in \mathbb{R}^{V \times V}$ is the cotangent Laplacian of S, $\mathbf{M} \in \mathbb{R}^{V \times V}$ is the respective mass matrix, and t_n is a parameter that controls the scale of diffusion. Accordingly, if we set $\mathbf{t} = \{t_n\}_{n=1}^{N}$ for all feature channels as learnable, it intuitively achieves channel-wise adaptive feature propagation over geodesic receptive fields flexibly determined in a data-driven fashion (from local to global). When combined with vertex-wise MLPs, such a simple operation is as expressive as complicated geodesic convolutions, which is theoretically guaranteed [5,18]. *More importantly*, compared with convolutions, this heat function-based learnable diffusion has two critical advantages: **1**) No need to pre-determine kernel sizes and largely reducing the number of learnable parameters; **2**) the principled foundation of the (continuous) heat function makes the diffusion results robust to the resolution of S and how it is meshed (which changes $\mathbf{L}$ and $\mathbf{M}$).

Although simple and powerful, the implementation of Eq.(2) is computationally infeasible due to large-scale inverse matrix. To address this challenge, a commonly used numerical solution is truncating diffusion to a *low-frequency* Laplacian basis of $\mathbf{L}$ and $\mathbf{M}$, i.e., the acceleration used in DiffusionNet [16]. Let $\boldsymbol{\lambda} = \{\lambda_i\}_{i=1}^{k} \in \mathbb{R}^{k \times 1}$ be the top-k eigenvalues, and $\boldsymbol{\Phi} = [\boldsymbol{\phi}_i]_{i=1}^{k} \in \mathbb{R}^{V \times k}$ denotes the respective orthogonal basis, satisfying $\mathbf{L}\boldsymbol{\phi}_i = \lambda_i \mathbf{M}\boldsymbol{\phi}_i$. Then, such a spectral-accelerated spatial diffusion is like:

$$H_{t_n}(\mathbf{U}[:,n]) \approx \boldsymbol{\Phi} exp(-t_n \boldsymbol{\lambda}) \odot (\boldsymbol{\Phi}^T \mathbf{M}\mathbf{U}[:,n]), \forall n = 1, \ldots, N, \tag{3}$$

where $\{\mathbf{L}, \mathbf{M}, \boldsymbol{\Phi}, \boldsymbol{\lambda}\}$ are mesh-specific and can be efficiently precomputed via standard package [13,15], and t_n is the per-channel learnable parameter.

However, it is worth noting that such a low-band approximation could lead to considerable errors in processing cortical surfaces, due to the loss of high-frequency information that is critical in describing highly folded structures. To explain this assumption, in Fig. 2, we show a representative example to approximate a subject in terms of different numbers of eigenvectors. As can be seen, even when k is as large as 1,000 (which means large computational burdens), there are still significant differences between the original and reconstructed surfaces.

Therefore, to make a more precise information aggregation and keep efficiency, we design a simple but intuitive strategy to realize full-band spectral acceleration of learnable spatial diffusion. Specifically, given the input feature $\mathbf{U}$,

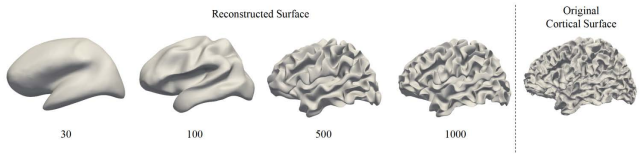

Fig. 2. An example to show the difference between original and reconstructed surfaces.

we first roughly separate the high-frequency part from its low-frequency part, such as $\mathbf{U}_h = \mathbf{U} - \mathbf{U}_l$ (where $\mathbf{U}_l = \boldsymbol{\Phi}\boldsymbol{\Phi}^T\mathbf{M}\mathbf{U}$). Then, a nonlinear mapping of $\mathbf{U}_h$ is performed as a residual complementary to help refine the lost details caused by spectral truncation. Our full-band solution is like:

$$H_{t_n}(\mathbf{U}[:,n]) \approx \boldsymbol{\Phi} exp(-t_n\boldsymbol{\lambda}) \odot (\boldsymbol{\Phi}^T\mathbf{M}\mathbf{U}[:,n]) + MLP_h(\mathbf{U}_h[:,n]), \forall n = 1,\ldots,N, \quad (4)$$

where $MLP_h(\cdot)$ (i.e., *high-freq MLP* in Fig. 1) is a simple three-layer MLP with ReLU after each layer. Due to such a straightforward design, we can properly truncate k while preserve fine-grained details in a lightweight fashion.

2.3 More Details

Overall, Cortex-Diffusion contains 4 FB-Diffusion blocks, each with 128 input and output channels (i.e., $N = 128$ in Fig. 1), respectively. To be lightweight, the design of the three-layer $MLP_h(\cdot)$ in each FB-SASD module draws inspiration from SqueezeNet [10], reducing the number of channels in the middle layer by half. We set $k = 200$ for spectral acceleration. The number of learnable parameters for the whole network is controlled as 0.49 MB. *Notably*, in both the training and test steps, our Cortex-Diffusion can flexibly process surfaces with varying sizes and mesh patterns, without the need to modify them to be the same.

3 Experiments

3.1 Experimental Setup

Datasets. We conducted comprehensive experiments to evaluate the efficacy of our methods on two public datasets: **1**) **Mindboggle** [11], an *adult* brain imaging dataset with 101 samples; **2**) **dHCP** [14], an *infant* brain imaging dataset with 558 samples. We used FreeSurfer [3] to reconstruct cortical surfaces in Mindboggle, and directly used the surface files released by dHCP project. Mindboggle has 32 ROIs, with the number of vertices ranging from 102k to 185k; dHCP has 17 ROIs, with the number of vertices ranging from 22k to 126k. The presented experimental results were quantified on the left hemispheric surfaces.

Comparison Methods. We compared our method with both the geometric deep networks based on spherical mapping, i.e., **1**) **Spherical U-Net** [19] and **2**) **SPHARM-Net** [7], and those can work directly on original surfaces, i.e., **3**) **SubdivNet** [9] and **4**) **DiffusionNet** [16]. Specifically, for Spherical U-Net and SPHARM-Net, we used FreeSurfer to perform spherical mapping, and then resample spherical surfaces onto the standard ico6 sphere. Following [19,7], mean curvature and sulcal depth were quantified as their methods' input. For SubdivNet, we used the features described in the original paper [9] as the input, and as required, we remeshed all cortical surfaces to have the same connectivity pattern by using the MAPS algorithm [12].

Table 1. Quantitative results obtained by different methods on Mindboggle.

Methods	Rotation Alignment	Input	Spherical Mapping	Params (MB)	Dice (%)
Spherical U-Net	No	cortical attributes	Yes	6.72	86.57 ± 2.63
SPHARM-Net				4.31	88.06 ± 1.99
SubdivNet		descriptor [9]	No	2.43	85.99 ± 2.21
DiffusionNet		xyz		0.47	87.12 ± 2.11
Ours				0.49	**88.18 ± 1.95**
Spherical U-Net	Yes	cortical attributes	Yes	6.72	87.96 ± 2.18
SPHARM-Net				4.31	88.00 ± 1.90
SubdivNet		descriptor [9]	No	2.43	86.53 ± 1.84
DiffusionNet		xyz		0.47	87.36 ± 1.96
Ours				0.49	**88.33 ± 1.90**

Implementation Details. We employed 5-fold cross-validation for performance quantification, i.e., three folds for training, one fold for validation, the remaining one fold for test, and iterated five times. We trained our Cortex-Diffusion by minimizing the vertex-wise cross-entropy loss. We used the ADAM optimizer with an initial learning rate of 0.001, with 0.5 decay if the loss on the validation set did not decrease for two consecutive epochs. In the test phase, we used the models with the highest Dice coefficient on the validation sets. We conducted all experiments on a PC with a NVIDIA RTX 3060 GPU. For fairness, all competing methods were implemented under the same training scheme. The performance was quantified in terms of **Dice**.

3.2 Results

Evaluation on adult cortical surfaces. The results obtained by different methods on the Mindboggle dataset are summarized in Table 1. This table delineates the performance outcomes under two distinct conditions categorized by "Rotation Alignment", i.e., the absence ("No") or presence ("Yes") of rigid alignment. This alignment pertains to whether the cortical surface analyses were conducted without or with a predefined standard orientation, respectively. Examination of Table 1 allows us to draw *three key conclusions*. *Primarily*, when compared with other approaches, our Cortex-Diffusion demonstrates superior accuracy in both scenarios. This is achieved through utilizing fundamental input data and foregoing the need for spherical mapping, thereby underscoring our method's precision and efficiency in cortical surface parcellation. *Additionally*, the comparison with DiffusionNet reveals that our method achieves a 1% improvement in Dice coefficient scores in both alignment conditions, while only necessitating a marginal increase in model complexity, specifically an additional 0.02 MB in learnable parameters. This increment underscores the effectiveness of our FB-SASD module's design. *Moreover*, our Cortex-Diffusion exhibits commendable performance consistency across both conditions, with a negligible reduction in Dice coefficient scores (0.15%) in the absence of alignment. This resilience against modest rotational variations among subjects is indicative of our method's robustness, achieving a level of performance on par with SPHARM-Net, which is explicitly designed to enhance rotational invariance.

As a supplementary, we also visualize the representative parcellations obtained by different methods in Fig. 3. As can been seen, our Cortex-Diffusion

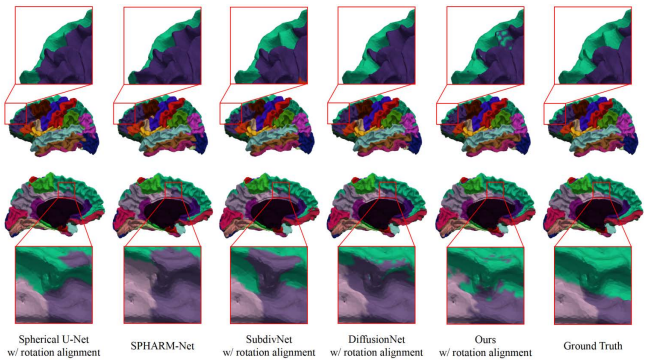

Fig. 3. Qualitative visualization of parcellations obtained by different models.

obtained more precise delineations especially at the challenging brain regions like the zoomed-in parts in Fig. 3. These visualization results further justify the efficacy of our method. Notably, here we only present the results in the case of rotation alignment. For more results without rotation alignment, please refer to the *supplemental material*.

Evaluation on infant cortical surfaces. The results obtained by different methods on the dHCP dataset are summarized in Table 2. Here, SubdivNet was removed from comparison, since its remeshing requirement is too time consuming on a PC for such a dataset with more than 500 subjects. From Table 2, we can see that our Cortex-Diffusion led to state-of-the-art performance on these infant cortical surfaces, which is consistent with the conclusion we draw in the adult case. Notably, the improvement by our method in Table 2 is of great value, considering that spherical mapping and cortical attribute quantification are themselves challenging tasks in infant brain image analyses.

Table 2. Quantitative results obtained by different methods on dHCP.

Methods	Input	Spherical Mapping	Params (MB)	Dice (%)
Spherical U-Net	cortical attributes	Yes	6.72	92.64 ± 1.39
SPHARM-Net			4.31	92.83 ± 1.25
DiffusionNet	xyz	No	0.46	93.00 ± 1.24
Ours			0.49	**93.84 ± 1.23**

Assessment of discretization-agnostic robustness. To evaluate the robustness of our method with respect to mesh resolutions and types (i.e., the discretization-agnostic property), we conducted a respective analysis on Mindboggle. Specifically, we used the QEM algorithm [4] implemented by the Trimesh toolkit [1] to simplify original high-resolution cortical surfaces, reducing the number of vertices per surface to a quarter. Subsequently, we evaluated competing methods across four distinct scenarios: training and testing on high-resolution meshes (**b**/**b**), training on high-resolution but testing on low-resolution meshes (**b**/**a**), training on low-resolution but testing on high-resolution meshes (**a**/**b**), and both training and testing on low-resolution meshes (**a**/**a**).

The results are summarized in Table 3. From Table 3, we can have two key observations. *First*, the performance of Spherical U-Net and SubdivNet drops a lot when the training and test settings are inconsistent (i.e., **b**/**a** and **a**/**b**). In contrast, SPHARM-Net, DiffusionNet, and our Cortex-Diffusion has relatively more stable accuracies across different situations thanks to the discretization-agnostically technical design. More specifically, SPHARM-Net leverages spherical harmonics [2], while our method and its backbone (DiffusionNet) rely on the heat diffusion functions, both are robust to the changes of vertical number and connectivity. *Second*, compared with SPHARM-Net and DiffusionNet, our Cortex-Diffusion has the best accuracy in the most challenging case of **a**/**b**, i.e., applying the networks trained on low-resolution surfaces to unseen high-resolution surfaces with changed connectivity patterns. It suggests the generalizability and practical usage of our method, considering that converting meshes to have the same patterns is a non-trivial task and may cause information losses.

Table 3. Assessment of the robustness of different methods concerning the changes between training and test settings. Here, x/y denotes training on x but testing on y; **b** and **a** represent original and down-sampled (i.e., remeshed) surfaces, respectively.

Methods	b/b	b/a	a/b	a/a
Spherical U-Net	86.57 ± 2.63	49.29 ± 7.13	27.81 ± 3.38	86.83 ± 2.52
SPHARM-Net	88.06 ± 1.99	**87.93 ± 2.01**	87.94 ± 1.91	**87.94 ± 1.92**
SubdivNet	85.99 ± 2.21	82.08 ± 2.29	78.66 ± 2.66	85.41 ± 2.02
DiffusionNet	87.12 ± 2.11	86.64 ± 2.05	87.12 ± 2.14	86.87 ± 2.06
Ours	**88.18 ± 1.95**	87.55 ± 2.01	**88.23 ± 1.98**	87.89 ± 1.91

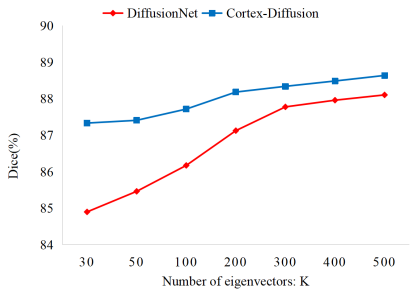

Fig. 4. Robustness of our Cortex-Diffusion method with respect to spectral-acceleration ratios.

Assessment of robustness to spectral-acceleration ratios. Our method combines low-frequency spectral acceleration with high-frequency MLP to approximate full-band spectral-accelerated spatial diffusion. The balance between the two parts is controlled by the parameter k, i.e., the number of eigenvectors in the Laplacian basis. To check the influence of k, we changed its values from 30 to 500, and compared our Cortex-Diffusion with its DiffusionNet backbone, with the results summarized in Fig. 4. As can be seen, the performance of DiffusionNet improves with the increase of k, as more information from larger frequency intervals is used for computing diffusion. However, an obvious marginal utility can be observed when k reaches 300. In contrast, our method consistently achieves better results across all situations. *Notably*, the performance of our method at

$k = 30$, already surpasses DiffusionNet at $k = 200$, further justifying the technical contribution of our FB-SASD module design.

4 Conclusion

In this paper, we have proposed an intuitive strategy to achieve full-band spectral-accelerated spatial diffusion learning, based on which a geometric deep network has been constructed for fully automated cortical surface parcellation in individual spaces. Our method leverages original surfaces' vertical coordinates as network input, producing accurate parcellation results without the need of time-consuming spherical operations. Both the evaluations on adult and infant cases have justified the good performance of our method as well as its robustness to the changes of mesh resolutions and connectivity.

Acknowledgments. This work was supported in part by STI 2030-Major Projects (No. 2022ZD0209000), and NSFC Grant (No. 62101430).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Dawson-Haggerty, M., et al.: trimesh. Python library (2019)
2. Driscoll, J.R., Healy, D.M.: Computing fourier transforms and convolutions on the 2-sphere. Advances in applied mathematics **15**(2), 202–250 (1994)
3. Fischl, B.: Freesurfer. Neuroimage **62**(2), 774–781 (2012)
4. Garland, M., Heckbert, P.S.: Surface simplification using quadric error metrics. In: Proceedings of the 24th annual conference on Computer graphics and interactive techniques. pp. 209–216 (1997)
5. Gasteiger, J., Weißenberger, S., Günnemann, S.: Diffusion improves graph learning. Advances in neural information processing systems **32** (2019)
6. Glasser, M.F., Sotiropoulos, S.N., Wilson, J.A., Coalson, T.S., Fischl, B., Andersson, J.L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J.R., et al.: The minimal preprocessing pipelines for the human connectome project. Neuroimage **80**, 105–124 (2013)
7. Ha, S., Lyu, I.: Spharm-net: spherical harmonics-based convolution for cortical parcellation. IEEE Transactions on Medical Imaging **41**(10), 2739–2751 (2022)
8. Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D.: Meshcnn: a network with an edge. ACM Transactions on Graphics (ToG) **38**(4), 1–12 (2019)
9. Hu, S.M., Liu, Z.N., Guo, M.H., Cai, J.X., Huang, J., Mu, T.J., Martin, R.R.: Subdivision-based mesh convolution networks. ACM Transactions on Graphics (TOG) **41**(3), 1–16 (2022)
10. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. arXiv preprint arXiv:1602.07360 (2016)
11. Klein, A., Tourville, J.: 101 labeled brain images and a consistent human cortical labeling protocol. Frontiers in neuroscience **6**, 171 (2012)
12. Lee, A.W., Sweldens, W., Schröder, P., Cowsar, L., Dobkin, D.: Maps: Multiresolution adaptive parameterization of surfaces. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 95–104 (1998)

13. Lehoucq, R.B., Sorensen, D.C., Yang, C.: ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods. SIAM (1998)
14. Makropoulos, A., Robinson, E.C., Schuh, A., Wright, R., Fitzgibbon, S., Bozek, J., Counsell, S.J., Steinweg, J., Vecchiato, K., Passerat-Palmbach, J., et al.: The developing human connectome project: A minimal processing pipeline for neonatal cortical surface reconstruction. Neuroimage **173**, 88–112 (2018)
15. Sharp, N.: potpourri3d (2021)
16. Sharp, N., Attaiki, S., Crane, K., Ovsjanikov, M.: Diffusionnet: Discretization agnostic learning on surfaces. ACM Transactions on Graphics (TOG) **41**(3), 1–16 (2022)
17. Wang, L., Wu, Z., Chen, L., Sun, Y., Lin, W., Li, G.: ibeat v2. 0: a multisite-applicable, deep learning-based pipeline for infant cerebral cortical surface reconstruction. Nature protocols **18**(5), 1488–1509 (2023)
18. Xu, B., Shen, H., Cao, Q., Cen, K., Cheng, X.: Graph convolutional networks using heat kernel for semi-supervised learning. arXiv preprint arXiv:2007.16002 (2020)
19. Zhao, F., Xia, S., Wu, Z., Duan, D., Wang, L., Lin, W., Gilmore, J.H., Shen, D., Li, G.: Spherical u-net on cortical surfaces: methods and applications. In: Information Processing in Medical Imaging: 26th International Conference, IPMI 2019, Hong Kong, China, June 2–7, 2019, Proceedings 26. pp. 855–866. Springer (2019)
20. Zöllei, L., Iglesias, J.E., Ou, Y., Grant, P.E., Fischl, B.: Infant freesurfer: An automated segmentation and surface extraction pipeline for t1-weighted neuroimaging data of infants 0–2 years. Neuroimage **218**, 116946 (2020)