Simultaneous Deformation and Individualized Parcellation of Cortical Surfaces

Anonymous WACV Applications Track submission

Paper ID 1510

Abstract

Cortical surface reconstruction and parcellation are essential for analyzing brain structure from structural MRI data. We present a novel architecture that performs parcellation directly on reconstructed cortical meshes, eliminating the need for spherical registration and drastically reducing processing time from hours to seconds compared to established methods like FreeSurfer and FastSurfer. Our approach employs a nearest neighbor technique to generate labels for surfaces with varying vertex counts and unknown correspondences relative to the ground truth mesh and annotations. We achieve competitive Dice scores in comparison to FastSurfer and demonstrate the algorithm's ability to generalize beyond training distribution meshes. We propose two training methodologies: one based on the nearest neighbor technique and another utilizing known FreeSurfer correspondences. Furthermore, our deformation model shows competitive performance in cortical surface reconstruction, achieving high accuracy in surface quality measures. We compare our model against several state-of-the-art Cortical Surface Reconstruction methods, highlighting its efficiency and effectiveness.

1. Introduction

Cortical surface reconstruction and parcellation are essential in neuroimaging research of the detailed architecture and functionalities of the brain [8, 12, 19]. Traditional methods often encounter difficulties in balancing computational efficiency with high geometric and topological accuracy. These hurdles present considerable challenges in both clinical practice and research. However, with the advent of deep learning and sophisticated mesh processing techniques, novel opportunities for enhancing these tasks have emerged. Yet, effectively integrating these progressive technologies into a streamlined pipeline still poses significant challenges.

In this work, we introduce a novel pipeline that integrates voxel-based methods, advanced mesh deformation, and graph neural network (GNN) architectures to perform cortical surface reconstruction and individualized subject parcellations. Our methodology significantly extends *CortexODE* [28] by using graph attention [45] to achieve receptive field previously obtained only through CNNs and Transformers in medical imaging, to introduce a novel framework for the generation and precise parcellation of cortical surfaces. This development sidesteps the need for spherical registration, streamlining the process of cortical mesh creation and parcellation.

For the deformation and parcellation process, we introduce an innovative architecture that combines the strengths of GNNs and Convolutional Neural Networks (CNNs). This dual approach is uniquely configured to generate precise deformations and individualized parcellations. The addition of the GNN allows for the utilization of larger receptive fields along the mesh. We compare our deformation architecture with a large number of leading Cortical Surface Reconstruction methods [7, 20, 24, 28, 29, 39], and we compare our parcellation pipeline against *FastSurfer* due to its robust efforts at enabling highly reproducible results [19].

Parcellation from structural MRI (sMRI) is underexplored yet crucial for advancing neuroimaging, as it provides detailed and individualized maps of brain structures. To date, none of the sub-minute cortical surface reconstruction methods operating in the sMRI coordinate system generate parcellations of the surfaces they produce, yet *FreeSurfer* and *FastSurfer* both generate such individualized surfaces and annotations [3, 7, 12, 19, 20, 24, 28, 29, 39].

We believe our methodology offers advancements in the following areas:

Personalized Medicine Individualized structural parcellations enable tailored diagnostic and therapeutic approaches, essential in the era of precision medicine. By acknowledging and addressing inter-individual variability in brain structure, parcellation facilitates more accurate and effective treatments for neurological conditions [25, 43, 48, 49].

Enhancement of Neuroimaging Techniques Architectural innovations are pivotal in enhancing the accuracy and

scalability of deep learning neuroimaging methods. The integration of CNNs, U-Nets, attention mechanisms, and GNNs has underpinned the success of existing approaches [18, 26, 40, 41, 46, 50]. Our architecture further capitalizes on these technologies to address challenges that were previously insurmountable.

Clinical Applications Detailed parcellations and cortical surfaces derived from sMRI are invaluable in clinical practice for diagnosing a range of neurological disorders including Alzheimer's, schizophrenia, and dementia [22, 36, 52, 53]. They assist in planning neurosurgical interventions and provide a deeper understanding of the anatomical bases of cognitive functions and disorders, thereby improving patient outcomes.

Research Implications Parcellation enhances our understanding of the relationship between brain anatomy and function [9, 11, 32, 34, 44]. It is instrumental in neuroscience research, offering insights into how structural variations across individuals influence cognitive and behavioral differences.

In summary, structural parcellation from sMRI is indispensable in neuroimaging, providing the foundation for personalized medicine, advancing imaging techniques, aiding clinical applications, and enriching neuroscience research.

Contributions Our contributions to the field of cortical surface reconstruction and parcellation techniques are as follows:

- **Propose and compare two individualized parcellation training methodologies for sMRI:** We train on both *FreeSurfer* surfaces and separately on deep learning-generated cortical surfaces, considering the generalizability of these techniques to meshes generated from disparate pipelines.
- **Novel neural network architecture:** We explore using Graph Attention [45] on existing deep learning-based Cortical Surface Reconstruction techniques to extend those methods into individualized deformation and parcellation, increasing their accuracy and speed in producing structural measures.
- **Novel measures for cortical surface comparison:** Many methods explore self-intersections, but none have examined the intersections of generated white and pial surfaces with each other. We propose this as a novel measure to evaluate surface quality.

2. Related Work

Cortical surface reconstruction using geometric deep learning is a burgeoning field that extends traditional deep learning techniques to model 3-dimensional objects and manifolds using triangular meshes [4–6]. Neural network architectures, loss functions, and pipelines have evolved to generate surfaces from magnetic resonance imaging (MRI) [3, 7, 20, 24, 28, 29, 39], effectively mirroring the characteristics of 3D data structures like sMRI data and meshes.

Meshes are fundamental in cortical surface reconstruction, representing the brain's White and Pial surfaces. These graph structures, with distinct areas, coordinates, and normal vectors, provide crucial insights into variations in cortical thickness, aiding in pinpointing aberrant brain regions [8, 12, 13, 20].

Existing frameworks predominantly use supervised learning, utilizing *FreeSurfer* to produce reference or "ground truth" meshes. These models aim to create "Genus 0" surfaces free of self-intersections or holes [7, 24, 39]. A Genus 0 surface, continuously deformable to a sphere, is crucial for accurate representation, especially in conformal mapping for group comparisons [17, 27, 30].

A significant challenge with *FreeSurfer* is its processing speed, taking around 6 hours to process a single patient's data using its most comprehensive approach [12, 20]. *FastSurfer* improves speed while preserving robustness, yet can still take from 45 minutes to 2 hours to run [19]. Fast deep learning-based cortical surface reconstruction techniques aim to outperform *FreeSurfer* in speed without compromising accuracy, yet many still exhibit topological anomalies, like self-intersecting surfaces. These deep learning frameworks are orders of magnitude faster than *FreeSurfer* and *FastSurfer*, leveraging GPU parallelization to match *FreeSurfer*'s output speed [3, 7, 20, 24, 28, 29, 39].

2.1. Voxel-Based Techniques

Voxel-based techniques typically use architectures like U-Net for segmentation, followed by topology correction and the marching cubes algorithm to extract the surface [14, 19]. For instance, *DeepCSR* employs an implicit surface framework that generates cortical surfaces using topology correction and marching cubes [7]. This method uses an occupancy field or a signed distance function (SDF) to represent the surface implicitly, applying marching cubes to extract the mesh. The generated mesh is then refined using Laplacian smoothing to enhance its quality and reduce artifacts.

2.2. Implicit Surface-Based Techniques

Implicit surface-based frameworks predict an occupancy field or a signed distance function (SDF), facilitating surface extraction using algorithms like marching cubes [7]. The SDF approach used by *DeepCSR* involves predicting the distance of each point in the MRI volume to the nearest surface point. This method benefits from the precise representation of the surface, allowing for smooth and detailed

reconstructions.

2.3. Mesh-Based Techniques

Mesh-based approaches typically start with an input template mesh and iteratively adjust it to closely match the voxel representation [3,20,24,29,39]. For example, *TopoFit* integrates image and graph convolutions to directly deform a Genus-0 mesh template into the detailed geometry of a subject-specific White matter surface [20]. This method leverages the strengths of Graph Neural Networks (GNN) and the U-Net architecture to process both image-derived and mesh-derived features, enabling precise estimation of deformation vectors necessary for accurate mesh adaptation.

CorticalFlow explores the use of ODE solvers in conjunction with neural networks to solve the cortical surface reconstruction problem [24]. The neural network predicts a diffeomorphic flow field, which allows for solving an initial value problem where the template mesh's vertices are the initial values. The flow field guides the deformation of the mesh via a smooth, homeomorphic mapping that ensures the structure remains intact and differentiable throughout the transformation process. *CorticalFlow++* enhances this approach by incorporating Runge Kutta integration and smooth templates, further refining the deformed surfaces and ensuring better topological properties [39].

CortexODE introduces a differential equation to represent the evolution of a cortical surface over time [28]. This framework segments the surface using a 3D U-Net, computes a signed distance function (SDF) for the segmentation map, applies Gaussian blur and topology correction, and then uses marching cubes to create an initial surface. This initial surface is further refined into the White matter surface using a deformation network.

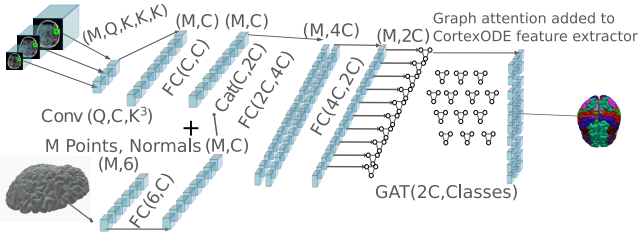

Figure 1. Neural network architecture of Our Parcellation (CSRVC) and Deformation (CSRF) networks. The architecture was extended from *CortexODE* [28].

2.4. Combined Approaches

Some frameworks combine implicit surface representations to generate initial surfaces, which are then refined into White and Pial surfaces. For instance, *Vox2Cortex* combines CNNs and GNNs to process sMRI data and generate detailed meshes of the cortex [3]. This framework starts by deforming a predefined mesh template according to features extracted from MRI scans, employing a CNN for image feature extraction and a GNN for mesh deformation. *Vox2Cortex* incorporates a novel loss function that prioritizes local curvature information, enhancing the accuracy of the mesh in representing the complex, folded structure of the cortex.

PialNN, another example, takes the White surface as input and produces a Pial surface, providing correspondences between the two surfaces. It employs cube sampling to gather features from the MRI at the coordinates indicated by the vertices in the White mesh surfaces, reducing the number of convolutions and focusing computational resources on relevant regions [29].

2.5. Parcellation

Parcellation has emerged as a particularly active area of research in functional MRI [33]. Moreover, studies using structural MRI have demonstrated the predictive power of structural parcellations in aging [23]. Structural parcellation of specific brain regions, such as the amygdala [42], anterior cingulate cortex [31], and corpus callosum [35], has enabled a deeper understanding of disease morphology. In the domain of cortical surface reconstruction, several researchers have attempted to segment the cortical surface [14, 15, 37, 47, 51].

3. Methodology

3.1. Deformation Pipeline (*CSRF*)

Our deformation model (*CSRF*) was created by extending *CortexODE*'s architecture to include a GNN using Graph Attention on its final layers instead of only an MLP [28]. We use an Euler solver to save training time compared to Runge-Kutta, with a step size of 0.1 as used in *CortexODE*. The graph attention layer provides a receptive field along the surface of the mesh, allowing deformation vectors to depend on the neighborhood as well as the current point. We selected our model, shown in Figure 1, based on the best validation scores found from searching through 2, 3, and 4 layers. We chose to search fewer layers because the ODE portion of the deformation, which is present in deformation but absent in our parcellation model, consumes memory and time. Thus, a smaller model is advantageous. We also train for 200 epochs instead of 400 epochs as we did with *CortexODE* due to time constraints and reasonably good validation loss values in fewer epochs.

3.2. Parcellation

For our parcellation results (CSRVC), we use the same architecture as for deformation (depicted in Figure 1), but

remove the ODE and apply two different training methodologies to explore the following questions:

- What are the most ideal results for this architecture when correspondence is known?
- Is it feasible to train with synthetic inputs from existing deep learning methods?
- Would these training methodologies generalize to meshes outside the training mesh distribution?

To address the first two questions, we developed the training methodologies illustrated in Figure 3. For the first question, we train on *FreeSurfer* surfaces as input and test on *FreeSurfer* surfaces as input with *FreeSurfer* generated annotations as ground truth labels from the DKTatlas, which are also available in *FastSurfer*'s extensive outputs. Calculating Dice scores is straightforward when correspondence is known, and the loss function for this training methodology is the Cross Entropy Loss commonly used in segmentation and classification. Our Dice scores exclude the -1 and 4 classes because these are non-cortex regions for that atlas.

For the second question, we train on *CortexODE* synthetic data as input meshes to annotate and map those vertices to *FreeSurfer* annotations using KDTrees and a nearest neighbor mapping. To exclude noise during training, we mask out vertices whose normals are greater than 60 degrees from their nearest neighbor normals in the ground truth *FreeSurfer* meshes.

To address the third question regarding the generalization of these training methodologies to meshes of a different distribution, we perform three tests. We test whether *FreeSurfer* trained models will properly annotate and predict on *CortexODE* and *CSRF* surfaces, and whether *CortexODE*-trained models will annotate and predict well on *FreeSurfer* and *CSRF* surfaces. When testing on *FreeSurfer* surfaces, the correspondence is known, allowing for a standard Dice calculation. When testing on either *CortexODE* or *CSRF* surfaces, we use the nearest neighbor mapping given by the KDTree to calculate an approximate Dice score.

For comparison, we use our approximate Dice calculation using KDTrees on *FastSurfer* generated surfaces on our test set [19].

Our parcellation model was selected based on the best validation results from a hyperparameter search over the number of GNN layers in the GAT, testing configurations with 2, 4, 6, 8, 10, and 12 layers. We train this model for 400 epochs since it is quick to train compared to ODE deformation models (*CSRF* was trained 200 epochs to save time).

3.3. Mesh Quality Measures

3.3.1 Distance to Ground Truth

Our research utilized a consistent test set across various frameworks, comprising 107 HCP patients. After applying *FreeSurfer*, the preprocessing pipelines for these frameworks entailed slightly divergent registration techniques. Despite the resulting outputs and ground truth meshes in spaces similar to *FreeSurfer*, we needed to ensure the spaces were nearly identical for precise distance calculations. To achieve this, we employed the iterative closest point (ICP) framework without necessitating alterations to the frameworks under consideration, alongside scaling and rotation adjustments (as required for *PialNN*). These procedures yielded a Linear Transformation matrix, which, when combined, produced a singular linear transformation matrix. This unified matrix facilitates the reversion of meshes to their original coordinate system.

Conceptually, this process involves three distinct meshes: the *FreeSurfer* ground truth mesh, denoted as $\mathbf{V}_{\text{gt}}$, the framework-specific representation of the ground truth mesh $\mathbf{V}_{\text{gt, framework}}$, and the model output mesh, denoted as $\widehat{\mathbf{V}}_{\text{pred}}$. We aim to derive a matrix $\mathbf{T}$ that aligns the framework-specific mesh $\mathbf{V}_{\text{gt, framework}}$ with the *FreeSurfer* ground truth $\mathbf{V}_{\text{gt}}$. Upon determining $\mathbf{T}$ through rotation, scaling, and ICP, we apply this matrix to both meshes. The distances are subsequently computed as follows:

$$\delta = d_{\text{dist}}(\mathbf{T}\widehat{\mathbf{V}}_{\text{pred}}, \mathbf{T}\mathbf{V}_{\text{gt, framework}}), \tag{1}$$

where $\mathbf{T}\mathbf{V}_{\text{gt, framework}}$ represents $\mathbf{V}_{\text{gt, framework}}$ transformed to approximately represent $\mathbf{V}_{\text{gt}}$ with minor distortions absent in the original mesh, and $\mathbf{T}\widehat{\mathbf{V}}_{\text{pred}}$ signifies the prediction transformed into the *FreeSurfer* ground truth space. Applying $\mathbf{T}$ to both $\mathbf{V}_{\text{gt, framework}}$ and $\widehat{\mathbf{V}}_{\text{pred}}$ ensures any minor errors in fitting are consistent in prediction and ground truth. Thus, for the iterative closest point, we map each mesh into the same space by selecting 500 landmark points and aligning them across the source (as produced by a model) and target (as produced by *FreeSurfer*), using the least squares sense via the ICP algorithm [2]. The ICP algorithm is performed after scaling and rotation, improving its performance and ensuring meshes occupy similar bounding box volumes as the reference. We calculated the distances between point clouds to the nearest point in the ground truth and visualized them in Figure 4—demonstrating Chamfer distance and Hausdorff distance. Chamfer distance is the average value of closest point distances between two distinct point clouds, while Hausdorff distance is the maximum distance of closest points in two point clouds. Providing the mean and the max via Chamfer and Hausdorff distances gives a better aggregate view of the meshes than either measure alone. We provide the distribution of Chamfer and Hausdorff distances in Figure 4.

3.3.2 Self Intersections

The assessment of self-intersections, akin to the inter-mesh collisions analysis in the next subsection, uses a simple and efficient algorithm. The only difference between self-intersection analysis in this section and multi-mesh White and Pial analysis in the next section is that the nearest point is excluded from consideration in finding self-intersections. For both self-intersections and the inter-mesh collision detection approach, we first construct a KDTree [1] representing centers of one of the meshes. Without loss of generality, consider this as the outer mesh. For self-intersections, the outer and inner mesh are the same. We then iterate through the triangles of the inner mesh, and for the center of each, we find the k nearest triangles from the outer mesh using the KDTree for greater efficiency. Intersections are only checked with these k nearest triangles, and a running counter maintains the tally of these intersections.

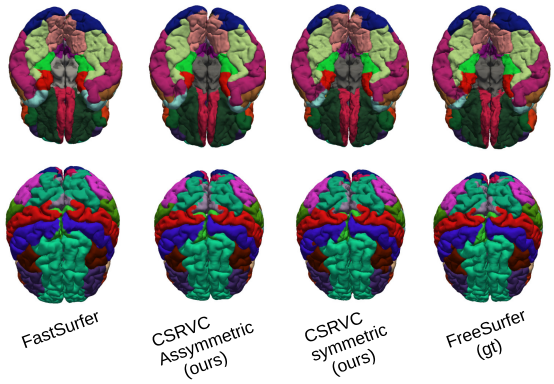

Figure 2. These are example parcellated surfaces predicted from FastSurfer, Our methods, and FreeSurfer. All methods produce outstanding parcellations.

Because self-intersections involve relationships independent of the coordinate system, no coordinate transformations were applied to predicted meshes before calculating self-intersections. Please note that the medial wall is excluded from topology correction with *FreeSurfer*'s default settings. To maintain a fair comparison, the medial wall predictions were not used in any evaluated models when calculating self-intersections. To remove the medial wall, we used the aparc.a2009s brain atlas and parcellation scheme proposed by Destrieux, Fischl, Dale, and Halgren to label the *FreeSurfer* ground truth mesh's ($\mathcal{M}_{\text{gt}}$ associated with $\mathbf{V}_{\text{gt}}$ mentioned in the previous section) medial wall [10]. Once we obtained the point cloud representing the medial wall, we then had to apply the inverse of $\mathbf{T}$ found previously to that point cloud, then removed faces in C_{pred} that were near the medial wall point cloud using k nearest neighbors algorithm. In our inspections, we discovered that the atlas fails to produce an excellent medial wall for some patients, so we exclude point clouds whose bounding box volumes are two standard deviations below the mean, which ensures that only reasonably large medial wall point clouds are used. Patients with small medial walls are excluded from all results in this section and the next.

3.3.3 Inter Mesh Collisions

Measuring inter-mesh collisions arose from our desire to see how the White and Pial meshes for each hemisphere interact. Much of the literature focuses on self-intersections of a single mesh, but fewer frameworks have investigated the interaction of the White and Pial mesh in terms of collisions and important quality for building numerical solvers for inverse problems such as boundary element methods [16].

We use the collision detection algorithm described in the previous self-intersections section to find the border triangles of this region.

3.4. Reference Deformation Model Training

We followed the instructions and defaults recommended for each framework in their respective readme files and uploaded configuration files for model training. Because the choice of architecture, scheduler, learning rate, and other design choices are coupled, we put the burden of reproducibility on the original authors for their instructions to result in Good Models. We generally used the highest resolution mesh settings for each framework that required templates. In *DeepCSR*, which wasn't template-based, we used a 256-resolution setting for our experiments and benchmarks since it was the highest stable resolution in our environment—likely due to memory usage at higher resolutions. For the frameworks that use ODE solvers, we used the default number of steps and solvers. For *CorticalFlow* and *CortexODE_euler*, we used an Euler solver. For *CorticalFlow++* and *CortexODE_rk4*, we used a Runge-Kutta 4 solver. *CortexODE* had a step size of .1 for both versions, which is the sweet spot for demonstrating the strengths of rk4 in their framework. *CorticalFlow* and *CorticalFlow++* defined their step parameter in terms of the number of steps, which we set to the default of 30 steps per solver. Our singularity training scripts reflect these settings.

3.5. Data and Code

The data used in this study are sourced from the Human Connectome Project (HCP). The specific datasets and subjects utilized from the HCP can be accessed through ConnectomeDB at https://db.humanconnectome.org/. For reproducibility and further research, the code will be released upon acceptance in the final version in order to be compliant with blind review.

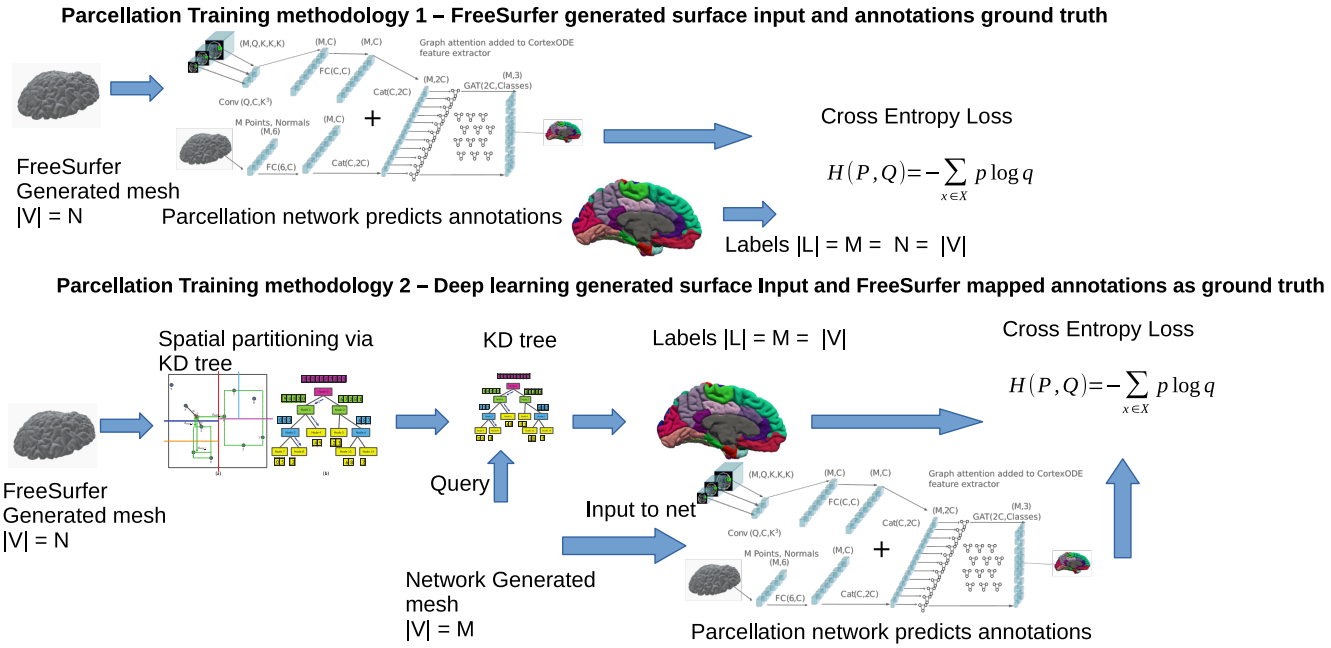

Figure 3. Neural network architecture of Our Parcellation (CSRVC) and Deformation (CSRF) networks. One can either use FreeSurfer or Deep learning networks to train for parcellation of Cortical Surfaces. KDTree in figure from [21]

For each framework, we started with HCP data processed by *FreeSurfer*, then applied any framework-specific preprocessing scripts.

3.6. Training Configuration

The system used for training is built on an x86_64 architecture using Ubuntu 20.04 OS. We allocated 4 Intel(R) Xeon(R) Platinum 8468 CPU cores for each surface type and framework. We requested 1 NVIDIA A100-SXM4-80GB GPU for each framework, with the NVIDIA driver version 535.104.12. The A100-SXM4-80GB model is equipped with 80GB of memory with CUDA 11 or above.

4. Results

4.1. Distance to Ground Truth

Much can be gleaned from the distance plot in Figure 4. Clearly, the winner for Pial surfaces is *CorticalFlow++*, yet there is a cluster of highly successful frameworks including *CSRF*, *CorticalFlow*, *CorticalFlow++*, *TopoFit*, and *CortexODE_rk4*. According to the *CortexODE* framework, using the Euler method would need more steps to be as successful as *CortexODE* with RK4, and this is apparent [28]. *DeepCSR* struggles to make competitive Pial surfaces but has almost competitive White surfaces. *Vox2Cortex* is less than ideal compared to state-of-the-art methods and even has catastrophic failures. *PialNN* makes commendable, but not great, Pial surfaces significantly improved in the author's make of *CortexODE* with RK4 solver.

4.2. Self Intersections

The self-intersection distributions in Figure 5 demonstrate the number of times the triangles in the mesh intersect other triangles in the meshes predicted for the four surfaces, left and right hemispheres White and Pial. *PialNN* has the highest topological error. *CorticalFlow* does not do well on Pial Surface Topology given the performances in distance measures, but *CorticalFlow++* significantly improves these numbers either through its use of better templates, Runge Kutta 4, or deforming the White surface into a Pial surface—presumably all of the above. White surfaces seem to be easier to create overall in this measure. *DeepCSR* is the champion of avoiding self-intersections due to its topology correction of the signed distance function and topology-preserving marching cubes. *CortexODE* has fantastic White surfaces, but middle-of-the-road Pial surfaces in terms of topology—an area that could be improved in future research. *FreeSurfer* does have some self-intersections, but not that many, given it is a log scale plot. *TopoFit* and *Vox2Cortex* perform fairly well overall on their White surface—excluding some of *Vox2Cortex*'s extreme outliers.

The number of collisions in a mesh can ruin some types of analysis, such as Geodesic distance calculations. Therefore, having measures of topological quality for these frameworks is vital information for anyone trying to determine if they can use deep learning in their analysis pipelines. Consequently, we provide these measures in Figure 5 and find that *DeepCSR* is exceptionally high quality

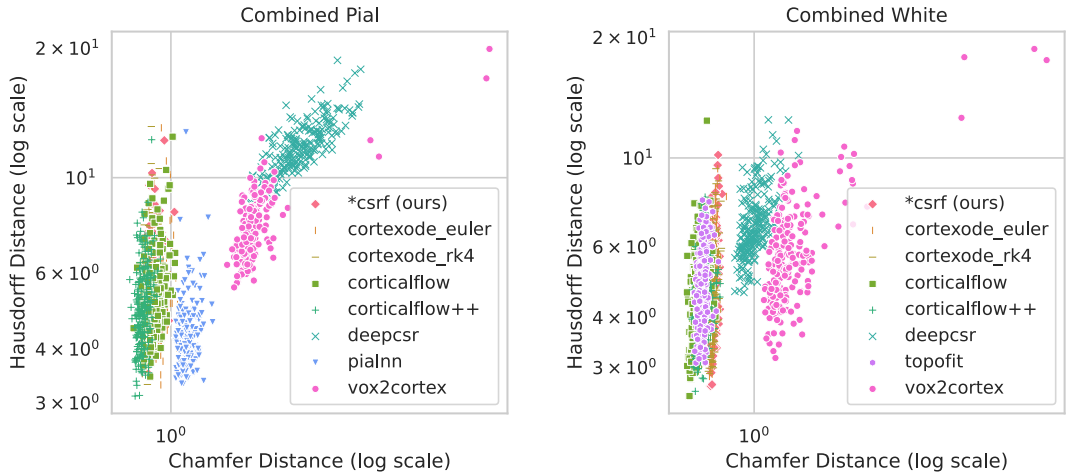

Figure 4. Distribution of distance measures for the test set predicted White and Pial surfaces for both hemispheres. Hausdorff distance represents the maximum distance of the closest points, while Chamfer distance represents the average distance of the closest points.

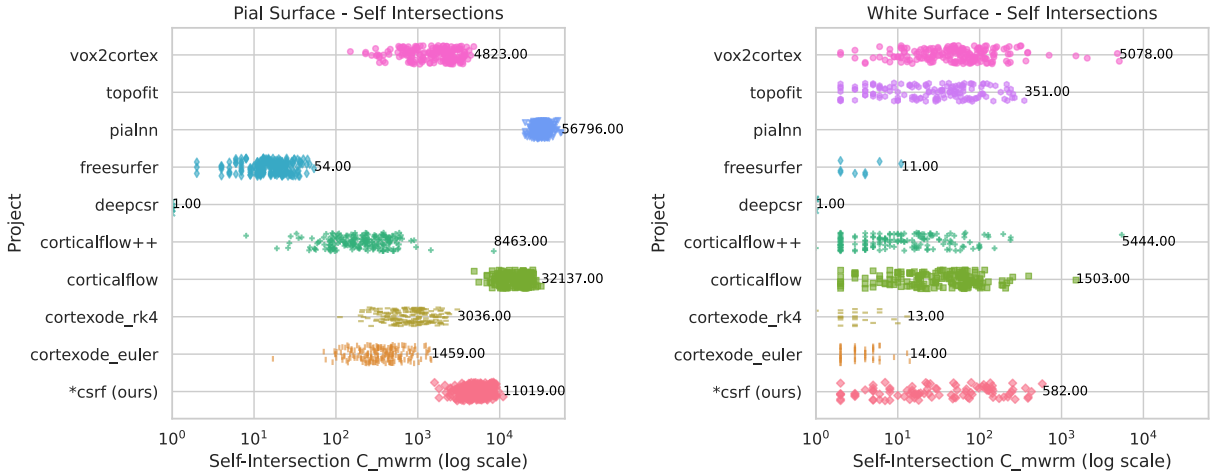

Figure 5. Collisions/Self-intersections of test set predicted surfaces (Higher is worse). These defects would affect geodesic distances in the cortex since they are distributed throughout the cortex. **FreeSurfer* was the ground truth, and its measurements are for reference.

in these measures. One could infer that topology correction combined with deep learning is best until faster solutions are created.

Additionally, training duration is likely a significant factor in these measures given we train *CSRF* for 200 epochs instead of 400 like *CortexODE*. *Vox2Cortex*'s training scheduler runs for roughly 100 epochs, so perhaps they could benefit from longer training as well.

4.3. Inter Mesh Collisions

For the frameworks that produce both surfaces in each hemisphere, we investigate whether the White and Pial surfaces intersect, which should not happen. These White-Pial intersections form the boundary where one surface protrudes into another. In these regions, cortical thickness must be better defined since these surfaces should never do this. The measures in Figure 6 count the number of triangles intersecting between the White and Pial surfaces in each hemisphere. We find that *CortexODE* does better in these measures than other frameworks. *FreeSurfer* is next best in these measures. We suspect that training duration is a major determinant here.

4.4. Parcellation

In addressing the questions posed in our methodology section for *CSRVC*, our results resoundingly indicate that our training methodology and architecture can generalize well to other meshes and training can be undertaken with

Table 1. Comparison of different cortical parcellation methods. Approximate Dice scores used nearest neighbor mappings for calculation and are useful to compare against *FastSurfer*. Exact measures are good for demonstrating the validity of the training methodologies we propose in Figure 3.

Dice Calculation	Training Input Surface	Testing Input Surface	Framework	GNN Layers (lh,rh)	Average Dice Score	Stdev Dice
exact	*CortexODE*	*FreeSurfer*	*CSRVC* (ours)	10,12	0.917	0.017
approximate	*CortexODE*	*CSRF* (ours)	*CSRVC* (ours)	10,12	0.880	0.018
approximate	*CortexODE*	*CortexODE*	*CSRVC* (ours)	10,12	0.875	0.018
exact	*FreeSurfer*	*FreeSurfer*	*CSRVC* (ours)	12,12	0.919	0.016
approximate	*FreeSurfer*	*CSRF* (ours)	*CSRVC* (ours)	12,12	0.879	0.017
approximate	*FreeSurfer*	*CortexODE*	*CSRVC* (ours)	12,12	0.874	0.017
-	-	-	*FastSurfer*	-	0.852	0.059

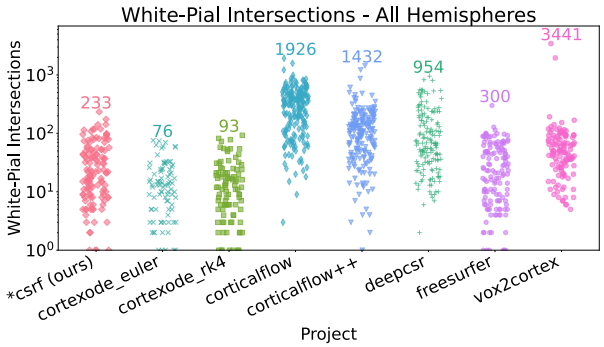

Figure 6. In cortical thickness measurements, White and Pial surfaces should not interweave, yet this figure demonstrates that White and Pial surfaces occasionally do interweave. We provide the number of triangle-face intersections of these two surfaces for both hemispheres. **FreeSurfer* was the ground truth, and its measurements are for reference.

synthetic data to generate good results using nearest neighbor mappings for the cross entropy loss function. This is especially apparent when we train using the nearest neighbor mapping with *CortexODE* surfaces, but test with *FreeSurfer* input surfaces and thus have access to an exact Dice measure at test time of 0.917. In terms of how well this architecture and methodology competes with *FreeSurfer*, we see that with our Approximate Dice score using the nearest neighbor mapping, all of our methodologies and tests beat *FastSurfer*, which also uses that Approximate Dice score due to asymmetries between the number of vertices in its surfaces and the ground truth *FreeSurfer* surfaces.

5. Conclusion

We observe that adding GAT to *CortexODE*'s network to produce *CSRF* and *CSRVC* is particularly effective for parcellation when the number of GNN layers is large, allowing for rapid training of deformation. Adding extra layers for deformation reduces the Chamfer distance, but may increase self-intersections without a smoothing loss term, similar to the findings in the more recent paper by the Vox2Cortex authors [38]. Additionally, training duration and the choice of ODE solver are crucial for achieving optimal results, with *CortexODE_rk4* and *CorticalFlow++* requiring multiple weeks to train.

Parcellation Dice results are influenced by whether strict correspondence can be found for annotations during training. However, training with approximate correspondence still achieves competitive results for individual parcellation, particularly when the Dice Calculation at test time is exact (not using nearest neighbor mappings) and thus more reliable. Our findings indicate that our training methodologies and parcellation architecture generalize well to meshes from different distributions.

In conclusion, the integration of GAT in our models demonstrates significant improvements in parcellation performance. Future work should explore the incorporation of smoothing loss terms to balance deformation accuracy and topological correctness. Additionally, optimizing training duration and solver configurations can further enhance model performance, ensuring robustness and generalizability across diverse datasets.

References

[1] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, sep 1975. 5

[2] PJ Besl and ND Mckay. A method for registration of 3-D shapes. *IEEE Trans. on Pattern Anal. and Machine Intelligence*, 1(4):23, 1992. 4

[3] Fabian Bongratz, Anne-Marie Rickmann, Sebastian Pölsterl, and Christian Wachinger. Vox2cortex: fast explicit reconstruction of cortical surfaces from 3d mri scans with geometric deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20773–20783, 2022. 1, 2, 3

[4] Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021. 2

[5] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017. 2

[6] Wenming Cao, Zhiyue Yan, Zhiquan He, and Zhihai He. A comprehensive survey on geometric deep learning. *IEEE Access*, 8:35929–35949, 2020. 2

[7] Rodrigo Santa Cruz, Leo Lebrat, Pierrick Bourgeat, Clinton Fookes, Jurgen Fripp, and Olivier Salvado. Deepcsr: A 3d deep learning approach for cortical surface reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 806–815, 2021. 1, 2

[8] Anders M Dale, Bruce Fischl, and Martin I Sereno. Cortical surface-based analysis: I. segmentation and surface reconstruction. *Neuroimage*, 9(2):179–194, 1999. 1, 2

[9] Marcel A de Reus and Martijn P Van den Heuvel. The parcellation-based connectome: limitations and extensions. *Neuroimage*, 80:397–404, 2013. 2

[10] Christophe Destrieux, Bruce Fischl, Anders Dale, and Eric Halgren. Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. *Neuroimage*, 53(1):1–15, 2010. 5

[11] Simon B Eickhoff, Bertrand Thirion, Gaël Varoquaux, and Danilo Bzdok. Connectivity-based parcellation: Critique and implications. *Human brain mapping*, 36(12):4771–4792, 2015. 2

[12] Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012. 1, 2

[13] Bruce Fischl, Martin I Sereno, and Anders M Dale. Cortical surface-based analysis: Ii: inflation, flattening, and a surface-based coordinate system. *Neuroimage*, 9(2):195–207, 1999. 2

[14] Karthik Gopinath, Christian Desrosiers, and Herve Lombaert. Segrecon: Learning joint brain surface reconstruction and segmentation from images. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part VII 24*, pages 650–659. Springer, 2021. 2, 3

[15] Karthik Gopinath, Christian Desrosiers, and Herve Lombaert. Learning joint surface reconstruction and segmentation, from brain images to cortical surface parcellation. *Medical Image Analysis*, 90:102974, 2023. 3

[16] Alexandre Gramfort, Martin Luessi, Eric Larson, Denis A. Engemann, Daniel Strohmeier, Christian Brodbeck, Roman Goj, Mainak Jas, Teon Brooks, Lauri Parkkonen, and Matti S. Hämäläinen. MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, 7(267):1–13, 2013. 5

[17] Xianfeng Gu, Yalin Wang, Tony F Chan, Paul M Thompson, and Shing-Tung Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE transactions on medical imaging*, 23(8):949–958, 2004. 2

[18] Kelei He, Chen Gan, Zhuoyuan Li, Islem Rekik, Zihao Yin, Wen Ji, Yang Gao, Qian Wang, Junfeng Zhang, and Dinggang Shen. Transformers in medical image analysis. *Intelligent Medicine*, 3(1):59–78, 2023. 2

[19] Leonie Henschel, Sailesh Conjeti, Santiago Estrada, Kersten Diers, Bruce Fischl, and Martin Reuter. Fastsurfer – a fast and accurate deep learning based neuroimaging pipeline. *NeuroImage*, 219:117012, 2020. 1, 2, 4

[20] Andrew Hoopes, Juan Eugenio Iglesias, Bruce Fischl, Douglas Greve, and Adrian V Dalca. Topofit: rapid reconstruction of topologically-correct cortical surfaces. *Proceedings of machine learning research*, 172:508, 2022. 1, 2, 3

[21] Linjia Hu, Saeid Nooshabadi, and Majid Ahmadi. Massively parallel kd-tree construction and nearest neighbor search algorithms. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2752–2755. IEEE, 2015. 6

[22] Maria Jalbrzikowski, Rebecca A Hayes, Stephen J Wood, Dorte Nordholm, Juan H Zhou, Paolo Fusar-Poli, Peter J Uhlhaas, Tsutomu Takahashi, Gisela Sugranyes, Yoo Bin Kwak, et al. Association of structural magnetic resonance imaging measures with psychosis onset in individuals at clinical high risk for developing psychosis: an enigma working group mega-analysis. *JAMA psychiatry*, 78(7):753–766, 2021. 2

[23] Huiting Jiang, Na Lu, Kewei Chen, Li Yao, Ke Li, Jiacai Zhang, and Xiaojuan Guo. Predicting brain age of healthy adults based on structural mri parcellation using convolutional neural networks. *Frontiers in neurology*, 10:1346, 2020. 3

[24] Leo Lebrat, Rodrigo Santa Cruz, Frederic de Gournay, Darren Fu, Pierrick Bourgeat, Jurgen Fripp, Clinton Fookes, and Olivier Salvado. Corticalflow: a diffeomorphic mesh transformer network for cortical surface reconstruction. *Advances in Neural Information Processing Systems*, 34:29491–29505, 2021. 1, 2, 3

[25] Chengyi Li, Shan Yu, and Yue Cui. Individual brain parcellation: Review of methods, validations and applications. *arXiv preprint arXiv:2407.00984*, 2024. 1

[26] Xiang Li, Minglei Li, Pengfei Yan, Guanyi Li, Yuchen Jiang, Hao Luo, and Shen Yin. Deep learning attention mechanism in medical image analysis: Basics and beyonds. *International Journal of Network Dynamics and Intelligence*, pages 93–116, 2023. 2

[27] Lok Ming Lui, Yalin Wang, Tony F Chan, and Paul Thompson. Landmark constrained genus zero surface conformal mapping and its application to brain mapping research. *Applied Numerical Mathematics*, 57(5-7):847–858, 2007. 2

[28] Qiang Ma, Liu Li, Emma C Robinson, Bernhard Kainz, Daniel Rueckert, and Amir Alansary. Cortexode: Learning cortical surface reconstruction by neural odes. *IEEE Transactions on Medical Imaging*, 2022. 1, 2, 3, 6

[29] Qiang Ma, Emma C Robinson, Bernhard Kainz, Daniel Rueckert, and Amir Alansary. Pialnn: a fast deep learning framework for cortical pial surface reconstruction. In *Machine Learning in Clinical Neuroimaging: 4th International Workshop, MLCN 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27, 2021, Proceedings 4*, pages 73–81. Springer International Publishing, 2021. 1, 2, 3

[30] Antonios Makropoulos, Emma C Robinson, Andreas Schuh, Robert Wright, Sean Fitzgibbon, Jelena Bozek, Serena J Counsell, Johannes Steinweg, Katy Vecchiato, Jonathan Passerat-Palmbach, et al. The developing human connectome project: A minimal processing pipeline for neonatal

cortical surface reconstruction. *Neuroimage*, 173:88–112, 2018. 2

[31] Laurie M McCormick, Steven Ziebell, Peggy Nopoulos, Martin Cassell, Nancy C Andreasen, and Michael Brumm. Anterior cingulate cortex: an mri-based parcellation method. *Neuroimage*, 32(3):1167–1175, 2006. 3

[32] Arnaud Messé. Parcellation influence on the connectivity-based structure–function relationship in the human brain. *Human Brain Mapping*, 41(5):1167–1180, 2020. 2

[33] Pantea Moghimi, Anh The Dang, Quan Do, Theoden I Netoff, Kelvin O Lim, and Gowtham Atluri. Evaluation of functional mri-based human brain parcellation: a review. *Journal of neurophysiology*, 128(1):197–217, 2022. 3

[34] Pantea Moghimi, Anh The Dang, Theoden I Netoff, Kelvin O Lim, and Gowtham Atluri. A review on mr based human brain parcellation methods. *arXiv preprint arXiv:2107.03475*, 2021. 2

[35] Hae-Jeong Park, Jae Jin Kim, Seung-Koo Lee, Jeong Ho Seok, Jiwon Chun, Dong Ik Kim, and Jong Doo Lee. Corpus callosal connection mapping using cortical gray matter parcellation and dt-mri. *Human brain mapping*, 29(5):503–516, 2008. 3

[36] VP Subramanyam Rallabandi, Ketki Tulpule, Mahanandeeshwar Gattu, Alzheimer's Disease Neuroimaging Initiative, et al. Automatic classification of cognitively normal, mild cognitive impairment and alzheimer's disease using structural mri analysis. *Informatics in Medicine Unlocked*, 18:100305, 2020. 2

[37] Anne-Marie Rickmann, Fabian Bongratz, Sebastian Pölsterl, Ignacio Sarasua, and Christian Wachinger. Joint reconstruction and parcellation of cortical surfaces. In *International Workshop on Machine Learning in Clinical Neuroimaging*, pages 3–12. Springer, 2022. 3

[38] Anne-Marie Rickmann, Fabian Bongratz, and Christian Wachinger. Vertex correspondence in cortical surface reconstruction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 318–327. Springer, 2023. 8

[39] Rodrigo Santa Cruz, Léo Lebrat, Darren Fu, Pierrick Bourgeat, Jurgen Fripp, Clinton Fookes, and Olivier Salvado. Corticalflow++: Boosting cortical surface reconstruction accuracy, regularity, and interoperability. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 496–505. Springer, 2022. 1, 2, 3

[40] Fahad Shamshad, Salman Khan, Syed Waqas Zamir, Muhammad Haris Khan, Munawar Hayat, Fahad Shahbaz Khan, and Huazhu Fu. Transformers in medical imaging: A survey. *Medical Image Analysis*, 88:102802, 2023. 2

[41] Satya P Singh, Lipo Wang, Sukrit Gupta, Haveesh Goli, Parasuraman Padmanabhan, and Balázs Gulyás. 3d deep learning on medical images: a review. *Sensors*, 20(18):5097, 2020. 2

[42] Eugenia Solano-Castiella, Andreas Schäfer, Enrico Reimer, Erik Türke, Thomas Pröger, Gabriele Lohmann, Robert Trampel, and Robert Turner. Parcellation of human amygdala in vivo using ultra high field structural mri. *Neuroimage*, 58(3):741–748, 2011. 3

[43] Si Jie Tang, Jonas Holle, Nicholas B Dadario, Olivia Lesslar, Charles Teo, Mark Ryan, Michael Sughrue, and Jacky T Yeung. Personalized, parcel-guided rtms for the treatment of major depressive disorder: Safety and proof of concept. *Brain and Behavior*, 13(11):e3268, 2023. 1

[44] Marc Tittgemeyer, Lionel Rigoux, and Thomas R Knösche. Cortical parcellation based on structural connectivity: A case for generative models. *Neuroimage*, 173:592–603, 2018. 2

[45] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 1, 2

[46] Jing Wang. Review of graph neural networks for medical image. *EAI Endorsed Transactions on e-Learning*, 9, 2023. 2

[47] Li Wang, Zhengwang Wu, Liangjun Chen, Yue Sun, Weili Lin, and Gang Li. ibeat v2. 0: a multisite-applicable, deep learning-based pipeline for infant cerebral cortical surface reconstruction. *Nature protocols*, 18(5):1488–1509, 2023. 3

[48] Jiang Zhang, Zhiwei Zhang, Hui Sun, Yingzi Ma, Jia Yang, Kexuan Chen, Xiaohui Yu, Tianwei Qin, Tianyu Zhao, Jingyue Zhang, et al. Personalized functional network mapping for autism spectrum disorder and attention-deficit/hyperactivity disorder. *Translational Psychiatry*, 14(1):92, 2024. 1

[49] Jiang Zhang, Tianyu Zhao, Jingyue Zhang, Zhiwei Zhang, Hongming Li, Bochao Cheng, Yajing Pang, Huawang Wu, and Jiaojian Wang. Prediction of childhood maltreatment and subtypes with personalized functional connectome of large-scale brain networks. *Human brain mapping*, 43(15):4710–4721, 2022. 1

[50] Lin Zhang, Yan Zhao, Tongtong Che, Shuyu Li, and Xiuying Wang. Graph neural networks for image-guided disease diagnosis: A review. *iRADIOLOGY*, 1(2):151–166, 2023. 2

[51] Fenqiang Zhao, Zhengwang Wu, Li Wang, Weili Lin, Shunren Xia, Gang Li, and UNC/UMN Baby Connectome Project Consortium. A deep network for joint registration and parcellation of cortical surfaces. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part IV 24*, pages 171–181. Springer, 2021. 3

[52] Xinxing Zhao, Candice Ke En Ang, U Rajendra Acharya, and Kang Hao Cheong. Application of artificial intelligence techniques for the detection of alzheimer's disease using structural mri images. *Biocybernetics and Biomedical Engineering*, 41(2):456–473, 2021. 2

[53] Wenyong Zhu, Liang Sun, Jiashuang Huang, Liangxiu Han, and Daoqiang Zhang. Dual attention multi-instance deep learning for alzheimer's disease diagnosis with structural mri. *IEEE Transactions on Medical Imaging*, 40(9):2354–2366, 2021. 2