Geometrical Learning of Brain Surface Data

by

Karthik GOPINATH

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE
TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
Ph.D.

MONTREAL, DECEMBER 2, 2021

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Herve Lombaert, Thesis Supervisor
Department of Software and IT Engineering, École de technologie supérieure

Mr. Christian Desrosiers, Co-Supervisor
Department of Software and IT Engineering, École de technologie supérieure

Mr. Marco Pedersoli, President of the Board of Examiners
Department of Electrical Engineering, École de technologie supérieure

Mr. Ismail Ben Ayed, Member of the Jury
Department of Electrical Engineering, École de technologie supérieure

Mr. Martin Reuter, External Examiner
Department of Radiology and Neurology, Harvard Medical School

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

To

Amma and Appa for everything

ACKNOWLEDGEMENTS

From the first video call to finishing the doctoral study, this journey has been the one I would cherish the most in my life. The contributions made in this thesis are only possible due to the overall support I received from my Ph.D. supervisors Prof. Herve Lombaert and Prof. Christian Desrosiers. I am very grateful to them for giving me this excellent opportunity. I feel very fortunate to have Prof. Herve Lombaert always available for discussion, encouraging and challenging me to explore new ideas. I have learned and benefited immensely from his guidance at École de Technologie superiéure (ÉTS). His work ethic and sincerity towards making our experience better always made working at the lab very conducive. Prof. Christian Desrosiers is equally the best supervisor any Ph.D. student could wish to have. His passion for research and curiosity motivated me to push and exceed my expectations. The advice and his attention to detail have taught me to be a better researcher. They have both been open, friendly, and assisted me in all aspects of this Ph.D., from welcoming me at the Montreal YUL Airport and fixing bureaucratic issues to sharing memes. I am immeasurably thankful to both of them for guiding me in this crucial period of my life and career. I would also like to express my sincere gratitude to my Masters by Research supervisor Prof. Jayanthi Sivaswamy for training me to navigate this captivating research world.

I am glad to have an opportunity to work with Prof. Ismail Ben Ayed. The conversations on new ideas and his knowledge in this research field are inspirational and thought-provoking. I have the immense pleasure to have known then post-doc and now Prof. Jose Dolz. From the initial hands-on coding sessions in Theano to exciting soccer matches, he has always been a helpful and constant support.

I am deeply thankful to all the jury members of my Ph.D. committee, Prof. Marco Pedersoli, Prof. Ismail Ben Ayed and Prof. Martin Reuter for accepting to read, evaluate and share thoughtful comments on my thesis.

At ETS, I spent most of my time working with colleagues in LIVIA and Shape Lab. Many thanks go to Kuldeep, Hoel, Saypra, Ziko, Le Thanh, Mathilde, Madhu, Akhil, and Sajjad. For all the memories, I also want to acknowledge the people of Shape Lab, Pierre, Benoit, and Sukesh. Names of many of these researchers don't appear in this manuscript. However, reading groups and talks in the labs have always been motivating and helpful.

On a personal front, I have to thank few friends who have played a vital role in this journey. I am sincerely grateful to have Raghav as a close friend, flatmate, and fellow Ph.D. candidate. He has always been available for me from technical discussion to stupid arguments. His immense breadth of knowledge has helped me in making complicated decisions seem easy. Additionally, I have also benefited from Sukesh and Chetan for their kindness and culinary skills. Finally, I also have to thank Samrudhdhi, Nehal, Rutuja, and Sumedh for making the weekends in Montreal unforgettable.

Last but not least, this thesis is dedicated to AMMA and APPA. Thank you very much for all your sacrifices and support.

Apprentissage géométrique de données surfaciques du cerveau

Karthik GOPINATH

RÉSUMÉ

Les corps cellulaires de neurones résident principalement dans le cortex cérébral. L'étude de cette fine surface, extrement pliée, est essentielle à la compréhension du fonctionnement du cerveau. Cependant, l'analyse des données surfaciques reste difficile dû à la grande variabilité de la géométrie corticale. Ignorer cette géométrie complexe pose un défi non-résolu pour l'analyse statistique des données surfacique. La plupart des méthodes conventionnelles reposent sur un processus heuristique à plusieurs étapes, telles que les simplifications géométriques et les inflations sphériques, nécessitant un temps de calcul considérable. L'objectif principal de cette thèse est de développer des cadriciels géométriques afin d'apprendre directement des données sur des surfaces corticales. Plus précisément, nous proposons d'abord un réseau de convolution de graphes spectraux pour effectuer une analyse de surface avec une application à la parcellation corticale. Ensuite, nous présentons une technique de regroupement adaptatif (*adaptive pooling*) pour des tâches de classification et de régression de données sur graphes dans le but de faciliter un apprentissage hiérarchique de données sur des surfaces multiples. Enfin, nous illustrons une approche conjointe de reconstruction et de segmentation de surfaces corticales afin d'exploiter les données de surfaces directement à partir d'un volume IRM. Cette thèse a donné lieu à trois publications dans des journaux et à cinq publications dans des conférences évaluées par les pairs. Les contributions individuelles de cette thèse sont présentées ci-dessous.

Dans notre premier objectif, nous présentons une nouvelle approche pour l'apprentissage et l'exploitation directe de données surfaciques sur de multiples domaines. L'apprentissage direct des données de surface via des convolutions de graphes fournit une nouvelle famille d'algorithmes rapides pour l'analyse de données corticales du cerveau. Cependant, la limite des approches actuelles réside principalement en leur incapacité à comparer les données de surface entre différents domaines de données. Les représentations surfaciques sont en effet incompatibles entre différentes géométries de cerveau. Nous tirons parti des récentes avancées en appariement de graphes spectraux pour faciliter le transfert de données surfacoques entre des domaines spectraux alignés. Cette nouvelle approche permet un apprentissage directement sur des surfaces à l'aide de bases spectrales compatibles. Un réseau de convolution de graphes spectraux peut ainsi exploiter des filtres spectraux sur les représentations intrinsèques de surfaces. Nous illustrons les avantages de cette approche par une application à la parcellation du cerveau. Nous validons l'algorithme sur 101 surfaces cérébrales étiquetées manuellement. Les améliorations de la parcellation indiquent une augmentation de 29% de la précision ainsi que des gains significatifs en capacité computationnelle par rapport aux méthodes conventionnelles.

Dans le deuxième objectif, nous proposons une nouvelle méthode d'apprentissage d'appariement sur graphes (*graph pooling*) afin de pouvoir aggréguer directement des données sur graphes provenant de multiples sur surfaces. La méthode présentée innove en apprenant une agrégation intrinsèque des nœuds d'un graphe basée sur l'analyse spectrale de graphe. Nous illustrons les

avantages de notre approche par des expériences exhaustives sur deux ensembles de données de référence. L'analyse par ablation présentée dans le chapitre illustre l'impact de divers facteurs sur notre méthode d'appariement. La flexibilité de la stratégie d'appariement est évaluée sur quatre tâches de prédiction différentes, à savoir la classification du sexe des sujets, la régression de la taille des régions corticales, la classification des stades de la maladie d'Alzheimer et la régression de l'âge du cerveau. Notre approche d'apprentissage d'appariement présente des améliorations allant de 7 à 11% par rapport à d'autres techniques d'appariement pour les réseaux convolutifs sur graphes, avec des résultats améliorant l'état de l'art de l'analyse de surfaces de cerveau.

Notre troisième objectif présente une stratégie antagoniste à l'adaptation de domaine non-supervisée afin d'apprendre des données de surface dans des domaines de graphes irréguliers. Cette nouvelle approche comprend un segmenteur qui utilise des couches de convolution de graphes pour permettre la parcellation de surfaces cérébrales à géométrie variable et un discriminateur qui prédit le domaine d'alignement des surfaces à partir de leur segmentation. L'entraînement antagoniste apprend une représentation invariante de l'alignement produisant des parcellations cohérentes pour des surfaces alignées différemment en trompant le discriminateur. En utilisant les surfaces cérébrales étiquetées manuellement de MindBoggle, le plus grand ensemble de données publiques de ce type, nous démontrons une amélioration de 2 à 13% de la moyenne de Dice par rapport à une stratégie d'apprentissage non contradictoire pour les surfaces cérébrales de test sans alignement ou alignées sur une référence différente des exemples sources.

Notre quatrième objectif final propose SegRecon, une méthode d'apprentissage profond intégrée de bout en bout pour reconstruire et segmenter conjointement les surfaces corticales directement à partir d'un volume IRM en une seule étape. Nous entraînons un réseau neuronal basé sur le volume pour prédire, pour chaque voxel, les distances signées à de multiples surfaces imbriquées et leur représentation sphérique correspondante dans l'espace atlasique. Ceci est, par exemple, utile pour reconstruire et segmenter conjointement l'interface matière blanche-matière grise et l'interface matière grise-CSF (pial). Nous évaluons les performances de notre méthode de reconstruction et de segmentation de surface à l'aide d'un ensemble complet d'expériences sur les jeux de données MindBoggle, ABIDE et OASIS. Notre erreur de reconstruction est inférieure à $0.52mm$ et $0,97mm$ en termes de distance de Hausdorff moyenne par rapport aux surfaces générées par FreeSurfer. De même, les résultats de la parcellation montrent une amélioration de plus de 4% de la moyenne de Dice par rapport à FreeSurfer, en plus d'une accélération drastique observée, passant de plusieurs heures à quelques secondes de calcul sur une station de bureau standard.

Le travail décrit dans cette thèse profitera aux études en neurosciences. De façon pratique, les algorithmes proposés aideront de manière significative les cliniciens à cibler une zone particulière du cerveau, utiles au développement de médicaments, et à prédire les précurseurs de l'atrophie corticale en exploitant explicitement la géométrie complexe du cortex cérébral et à isoler la géométrie discriminante liée à la maladie d'Alzheimer. De plus, cette thèse peut aider à l'extraction rapide et précise de surfaces et à leur parcellation directement à partir de volumes IRM structurels. Ce travail permettra également de réduire la charge financière des

patients en fournissant des outils algorithmiques pour la recherche thérapeutique. D'une manière générale, ce travail sera utilisé pour trouver de nouveaux biomarqueurs basés sur la géométrie pour mieux détecter la maladie d'Alzheimer et pour aider à la compréhension d'autres troubles neurologiques.

Mots-clés: Apprentissage profond géométrique, Parcellation corticale, Regroupement sur graphes, Segmentation et reconstruction de surface

Geometrical Learning of Brain Surface Data

Karthik GOPINATH

ABSTRACT

Neuronal cell bodies primarily reside in the cerebral cortex. The study of this thin and highly convoluted surface is essential for understanding how the brain works. However, the analysis of surface data is challenging due to the high variability of the cortical geometry. Ignoring this complex geometry poses an unsolved challenge in the statistical analysis of surface data. Most conventional methods rely on a heuristic multi-step process, such as geometrical simplifications and spherical inflations, requiring a considerable computational time. The main objective of this thesis is to develop geometrical frameworks to learn directly on cortical surfaces. Specifically, we first propose a spectral graph convolution network to perform surface analysis applied to cortical parcellation. Next, we present an adaptive pooling technique for surface classification and regression tasks to facilitate a hierarchical learning of multiple surface data. Finally, we illustrate a joint cortical surface reconstruction and segmentation approach to work directly from the MRI volume. This thesis has resulted in three journals and five peer-reviewed conference publications. The individual contributions of this thesis are presented below.

In our first objective, we present a novel approach for learning and exploiting surface data directly across multiple surface domains. Direct learning of surface data via graph convolutions provide a new family of fast algorithms for processing brain surfaces. However, the current limitation of existing state-of-the-art approaches is their inability to compare surface data across different surface domains. Surface representations are indeed incompatible between brain geometries. We leverage the recent advances in spectral graph matching to transfer surface data across aligned spectral domains. This novel approach enables a direct learning of surface data across compatible surface bases. A spectral graph convolution network exploits spectral filters over intrinsic representations of surface neighborhoods. We illustrate the benefits of this approach with an application to brain parcellation. We validate the algorithm over 101 manually labeled brain surfaces. The improvements in parcellation reveal a 29% increase in accuracy with drastic speed gains over conventional methods.

In the second objective, we propose a new learnable graph pooling method for processing multiple surface-valued data to output subject-based information. The presented method innovates by learning an intrinsic aggregation of graph nodes based on graph spectral embedding. We illustrate the advantages of our approach with in-depth experiments on two large-scale benchmark datasets. The ablation study in the chapter illustrates the impact of various factors affecting our learnable pooling method. The flexibility of the pooling strategy is evaluated on four different prediction tasks, namely, subject-sex classification, regression of cortical region sizes, classification of Alzheimer's disease stages, and brain age regression. Our learnable pooling approach demonstrates improvements ranging from 7% to 11% compared to other pooling techniques for graph convolutional networks, with results improving the state-of-the-art in brain surface analysis.

Our third objective presents an adversarial training strategy for unsupervised domain adaptation to learn surface data across inconsistent graph domains. This novel approach comprises of a segmentator that uses graph convolution layers to enable parcellation across brain surfaces of varying geometry and a discriminator that predicts the alignment-domain of surfaces from their segmentation. The adversarial training learns an alignment-invariant representation that yields consistent parcellations for differently aligned surfaces by fooling the discriminator. Using manually-labeled brain surface from MindBoggle, the largest publicly available dataset of this kind, we demonstrate a 2%–13% improvement in mean Dice over a non-adversarial training strategy for test brain surfaces with no alignment or aligned on a different reference than source examples.

Our fourth final objective proposes SegRecon, an integrated end-to-end deep learning method to jointly reconstruct and segment cortical surfaces directly from an MRI volume in one single step. We train a volume-based neural network to predict, for each voxel, the signed distances to multiple nested surfaces and their corresponding spherical representation in atlas space. This is, for instance, useful for jointly reconstructing and segmenting the white-to-grey-matter interface and the grey-matter-to-CSF (pial) surface. We evaluate the performance of our surface reconstruction and segmentation method with a comprehensive set of experiments on the MindBoggle, ABIDE and OASIS datasets. Our reconstruction error is found to be less than 0.52 mm and 0.97 mm in terms of average Hausdorff distance to the FreeSurfer generated surfaces. Likewise, the parcellation results show over 4% improvements in average Dice with respect to FreeSurfer, in addition to an observed drastic speed-up from hours to seconds of computation on a standard desktop station.

The work described in this thesis benefits neuroscience studies. Practically, the proposed algorithms will significantly assist clinicians in targeting any particular area of the brain for drug planning and in early prediction of cortical atrophy using the geometry of the complex folding of the cortex and isolate the discriminating geometry linked with the Alzheimer's disease. Additionally, this thesis can help in fast and accurate surface extraction and parcellation from structural MRI volumes. This work will also reduce financial burdens on patients by providing algorithmic tools for therapeutic research to aid clinicians. In general, the work would be used to find new geometry-based biomarkers for the early detection of the Alzheimer's disease and assist the understanding of other neurological disorders.

Keywords: Geometric deep learning, Cortical surface parcellation, Graph pooling, Surface reconstruction segmentation

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND UNITS OF MEASUREMENTS

$\mathbf{A}$	Weighted adjacency matrix
$\mathcal{X}$	Training set composed of 3D volumes
$\mathcal{X}_{\text{src}}$	Set of source graphs
$\mathcal{X}_{\text{tgt}}$	Set of unlabeled domain graphs
$\mathbf{D}$	Diagonal degree matrix
D	Discriminator GCN
$\mathcal{E}$	Edge set of graph $\mathcal{G}$
$\mathcal{G}$	Graph defined with node set $\mathcal{V}$, such that $N = \|\mathcal{V}\|$, and edge set $\mathcal{E}$.
$\mathbf{I}$	Identity matrix
X	3D MRI volume
$\mathbf{L}$	Normalized graph Laplacian
$\mathbf{U\Lambda U}^\top$	Eigendecomposition of the normalized Laplacian matrix $\mathbf{L}$
$\mathbb{1}$	Indicator function
S^W	White cortical surface
$\varphi_{ij}(\mathbf{\Theta}_k)$	Symmetric kernel with parameters $\mathbf{\Theta}_k$
$\widehat{\mathbf{U}}$	Normalized spectral coordinates of nodes as the rows of matrix
$\widetilde{\mathbf{U}}$	Aligned spectral coordinates as the rows of matrix

LIST OF ABREVIATIONS

AAD	Absolute Average Distance
AMI	Adjusted Mutual Information
Adv-GCN	Adversarial Graph Convolution Network
AD	Alzheimer’S Disease
ASD	Autism Spectrum Disorder
BW	Binary Mask
CD	Chamfer Distance
CNS	Central Nervous System
CSF	Cerebro Spinal Fluid
CRF	Conditional Random Field
CNNs	Convolutional Neural Networks
CE	Cross-Entropy
DKT	Desikan-Killiany-Tourville
dMRI	Diffusion Magnetic Resonance Imaging
FC	Fully-Connected
FCN	Fully-Convolutional Network
fMRI	Functional Magnetic Resonance Imaging
FRQNT	Fonds De Recherche Du Québec Nature Et Technologies

GANs	Generative Adversarial Networks
GCN	Graph Convolution Network
HD	Hausdorff Distance
HCP	Human Connectome Project
IPMI	International Conference On Information Processing In Medical Imaging
ICP	Iterative Closest Point
LReLU	Leaky Rectified Linear Unit
MRI	Magnetic Resonance Imaging
Meg	Magnetoencephalography
MRF	Markov Random Field
MAE	Mean Absolute Error
MSE	Mean Squared Error
MedIA	Medical Image Analysis
MICCAI	Medical Image Computing And Computer Assisted Intervention
MCI	Mild Cognitive Impairment
NC	Control Subjects
NSERC	Natural Sciences And Engineering Research Council Of Canada
PNS	Peripheral Nervous System
RF	Random Forest
ReLU	Rectified Linear Unit

Seg-GCN	Segmentation Graph Convolution Network
SD	Signed Distance
SRF	Spectral Random Forest
SGD	Stochastic Gradient Descent
sMRI	Structural Magnetic Resonance Imaging
UDA	Unsupervised Domain Adaptation

INTRODUCTION

The brain is the largest processing unit of the nervous system. The overall structure of the brain is most common across the mammal species (Budday, Steinmann III & Kuhl, 2015). The Figure 0.1 shows a comparison between different mammal species illustrating the size difference. It is responsible for voluntary and involuntary actions, perception, thinking, emotions, planning, behavior, and memory. Due to the complexity of functions the brain performs, understanding the working of the brain is vital for the neuroscience community. Next, a brief description of the general anatomy of the brain and its surface is provided.

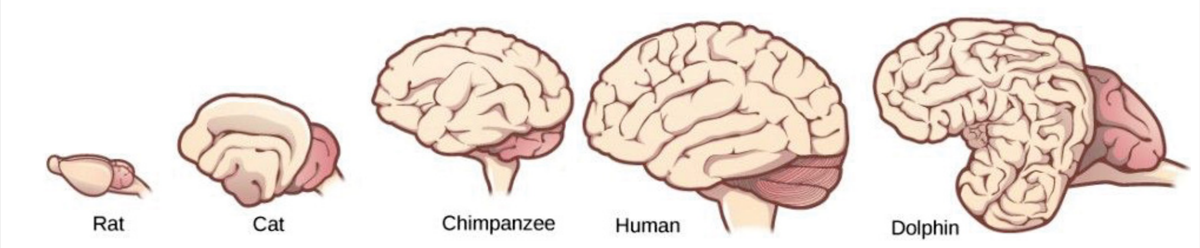


Figure 0.1 Different species brain comparison – The complexity and structural difference of brain across mammals and vertebrae species. Taken from Molnar *et al.* (2015, p. 687)

0.1 Anatomy of the Brain

The brain comprises three primary tissues at a cellular level, namely gray matter, white matter, and cerebrospinal fluid (CSF) (Nolte, 1993; Saladin & Miller, 1998). The left figure in Fig. 0.2 shows these three tissue types of the brain. Grey matter is the processing part of neurons in the living brain. White matter consists of bundles of myelinated axons which link various parts of the brain to information exchanges. CSF is the fluid present in the ventricles acting as a cushion to the brain, providing essential mechanical and immunological protection.

The brain at a higher level is composed of three central units called the brainstem, cerebellum, and cerebrum (Molnar *et al.*, 2015). On the right of Fig. 0.2, shows the three central units of

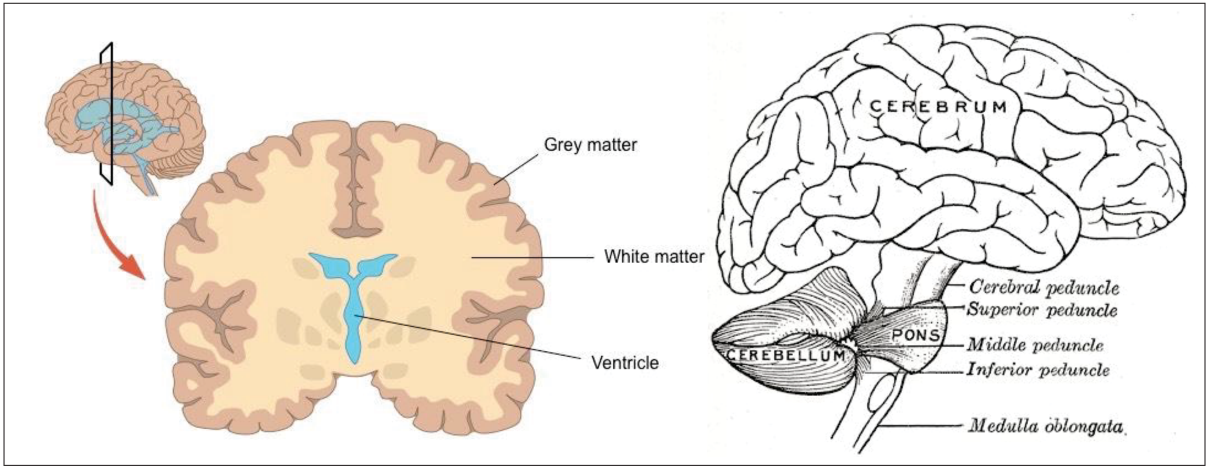


Figure 0.2 Anatomy of the brain – Left: The three main tissue types of healthy human brain. Taken from Cornell (2016); Right: The brain stem, cerebellum and the Cerebrum of the human brain. Taken from Gray (1918)

the brain. The brainstem is a distinctive center that connects the spinal cord to the brain. The primary function of the brainstem is linked to automatic, involuntary actions like heart rate, breathing, digestion, and sleep cycles. The cerebellum attached to the brainstem is linked with motor and muscle movements, maintaining balance and posture. Above the cerebellum is the largest part of the brain called the cerebrum. The cerebrum, also known as the cerebral cortex, is made of the right and left hemispheres. Billions of neurons reside on the surface, performing higher functions like vision, touch, hearing, speech learning, and emotions.

0.1.1 Brain Surface – Cerebral Cortex

The surface of the brain or the cerebral cortex is a remarkably complex organ functionally and structurally. It is particularly convoluted with numerous folding patterns. The ridges and the grooves of the cortical surfaces are referred to as gyri and sulci, respectively. These folding patterns assist this structural division into two hemispheres connected by the corpus callosum (Molnar *et al.*, 2015). This thick band of neural fibers contains millions of axons responsible for inter-hemispheric communication.

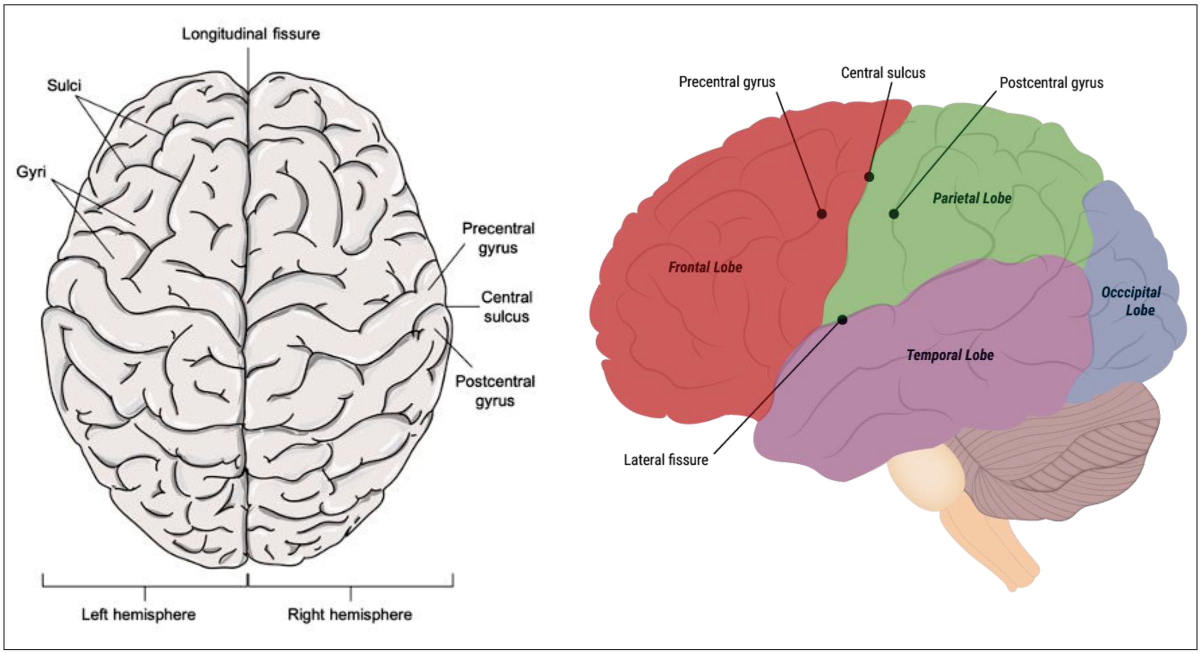


Figure 0.3 Gyri, sucli and lobes of the brain – Left: The brain cortex depicting the gyri, sulci and the hemispheres. Taken from Butte (2021); Right: The main four areas or lobes of the brain cortex. Taken from Fejtek (2019)

The surface area of the brain is increased by facilitating numerous foldings to fit the skull of the head. This increase in the folding of the surface enabling higher functions (Shipp, 2007). During the initial development, the cortex starts as a smooth surface and develops complex convolution patterns(Razavi, Zhang, Liu & Wang, 2015). The primary folding formed at the central sulcus is similar across a population, with the secondary and tertiary folding patterns developing unique signatures (Tallinen, Chung, Rousseau, Girard, Lefèvre & Mahadevan, 2016).

Areas of the Cerebral Cortex

Each hemisphere of the cerebral cortex has four different lobes based on distinctive fissures. The frontal, parietal, temporal, and occipital lobes control specific neural activity on the cortical surface. The Frontal lobe is the largest region in the front of the head is responsible for problem-solving, behavior, emotions, and planning. The Parietal lobe in the middle-top part of the brain is responsible for sensations like touch, pain, and pressure. In the back of the brain, the occipital lobe is involved in assessing visual and spatial information. The temporal lobes at the sides are responsible for sound, smell, and memory.

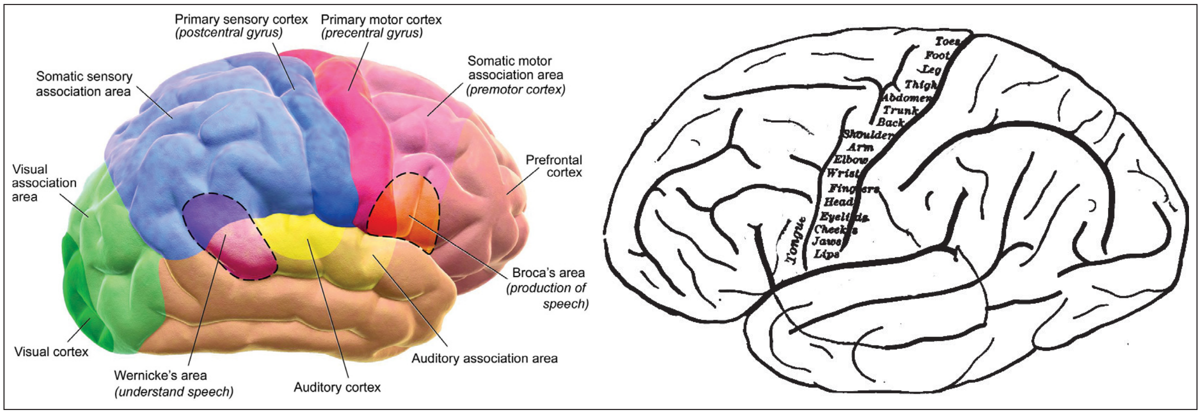


Figure 0.4 Areas of the cerebral cortex – Left: The main functional areas of the cerebral cortex. Taken from Bruce Blaus (2014); Right: Primary motor cortex represented by areas associated to different body parts along the central sulcus. Taken from Wikimedia Commons (2011)

The cerebral cortex of the brain can also be divided further into different functional areas (Fischl, van der Kouwe, Destrieux, Halgren, Segonne, Salat, Busa, Seidman, Goldstein, Kennedy, Caviness, Makris, Rosen & Dale, 2004a; Klein, Ghosh, Bao, Giard, Häme, Stavsky, Lee, Rossa, Reuter, Chaibub Neto & Keshavan, 2017). In particular, a strip of the region called the primary motor cortex along the side of the brain is responsible for voluntary motor actions like handshakes and smiling. Fig. 0.4 (right) shows the regions in the central sulcus associated with the corresponding body parts. Neuroscientists have been studying the primary cortical area of the brain to divide them into anatomically and functionally relevant regions. In literature, some works divide the brain roughly from 30 to 200 regions for each hemisphere (Klein *et al.*, 2017).

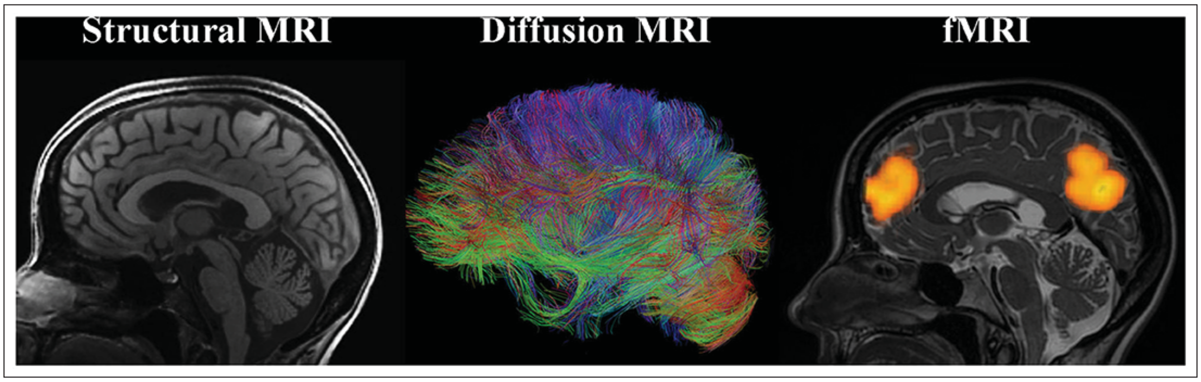


Figure 0.5 Brain imaging techniques – The different MRI imaging techniques used for analysing human brain. Taken from Tymofiyeva & Gaschler (2021, p. 3)

0.2 Imaging the Brain

In the last few decades, neuroimaging has become an essential tool to visualize non-invasively the human brain (Cacioppo, Berntson & Nusbaum, 2008; Geffroy, Rivière, Denghien, Souedet, Laguitton & Cointepas, 2011). This technology has attracted people from different research fields (Epstein, Stern & Silbersweig, 2001) to attempt at understanding the brain. Various aspects of the mind are studied using imaging modalities such as magnetic resonance imaging (MRI) and magnetoencephalography (MEG). Different MRI-related techniques namely structural MRI (sMRI), functional MRI (fMRI), and diffusion MRI (dMRI) are used to obtain information about the structure and function of the brain. Visualization of the brain at one cubic millimeter resolution is possible with the advancements in the structural MRI techniques (Liang & Lauterbur, 2000). Fig 0.5 shows images obtained from different MRI techniques. These images are ideal for studying brain structures along with abnormalities like tumors or lesions. Functional areas of the brain associated with actions are analyzed using data obtained from fMRI. fMRI measures brain activity by detecting changes in the blood oxygen (DeYoe, Bandettini, Neitz, Miller & Winans, 1994; Ogawa, Lee, Kay & Tank, 1990). On the other hand, dMRI allows the inferring white matter connectivity between different anatomical regions of the brain (Le Bihan, Mangin, Poupon, Clark, Pappata, Molko & Chabriat, 2001; Merboldt, Hanicke & Frahm, 1985).

0.3 Challenges and Motivation

Neuroimage analysis consists of studying functional and anatomical information over the brain geometry. The thin outer layer of the brain cerebrum is of particular interest due to its vital role in cognition, vision, and perception. Neurological disorders affect nearly 1 in 6 adults in the world, causing financial burdens to millions of lives (DiLuca & Olesen, 2014; Ernst & Hay, 1994; Rice, 1999). Every minute someone in North America, including Quebec, develops Alzheimer's disease (AD). AD is characterized by the loss of cognitive function due to the impairment of the neuronal connections (Arbabshirani, Plis, Sui & Calhoun, 2017). The concern on AD is that it is challenging to identify its early stages, with currently no therapies to stop or cure the disease. Schizophrenia is another disorder that affects about 1% of the total population (Bhugra, 2005). This disease is usually characterized by cognitive problems, and misinterpretation of reality (Heinrichs & Zakzanis, 1998). Major depressive disorder, as the name suggests, is related to a lack of interest in activities traditionally enjoyed by sufferers, including pervasive low mood, self-esteem (Kessler, Berglund, Demler, Jin, Koretz, Merikangas, Rush, Walters & Wang, 2003; Lewinsohn, Duncan, Stanton & Hautzinger, 1986). By the end of 2030, depression will be ranked second for the financial burden on society, right after heart disease (Kessler, McGonagle, Zhao, Nelson, Hughes, Eshleman, Wittchen & Kendler, 1994). Fig. 0.6 illustrates how diseases such as Alzheimer's can affect the brain. Structural MRI is often used to study the brain atrophy associated with AD and other disorders or diseases (Buffalo, Movshon & Wurtz, 2019). A learning framework operating directly on the surface can aid the identification of biomarkers for AD, drug planning, and surgical interventions.

Limitations of the conventional methods make it hard to learn directly over the surface data. Current popular approaches, for instance, FreeSurfer (Fischl *et al.*, 1999) typically rely on geometrical simplifications, such as spherical inflation and slow mesh deformations and takes around 3 hours for surface analysis. Deep learning methods on medical imaging data have

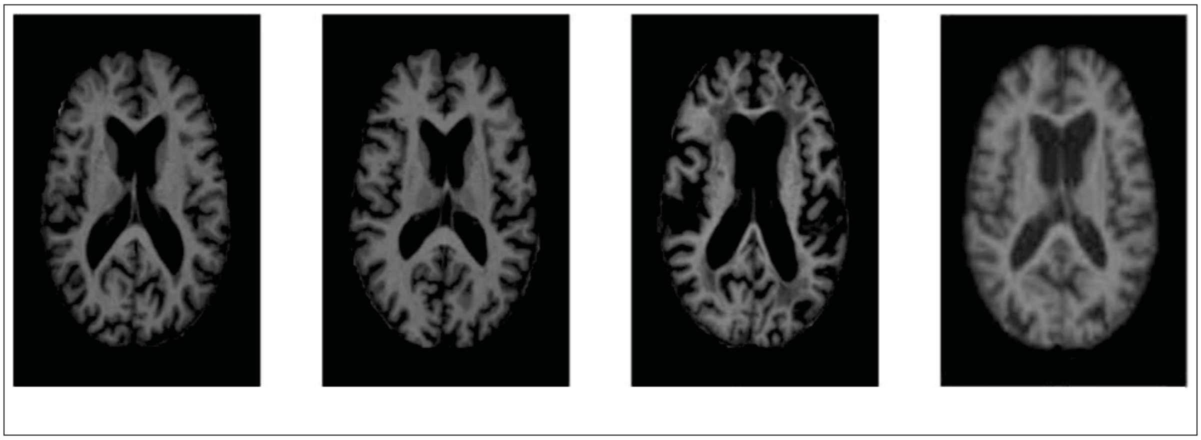


Figure 0.6 Normal brain vs diseased brain stages – An illustration of normal brain vs stages of diseased brain. Adapted from Islam & Zhang (2018, p. 2)

shown drastic improvements in classification and segmentation problems. Convolutional Neural Networks (CNNs) can offer a drastic speed advantage over traditional surface-based methods. Fundamentally, current CNNs exploit spatial information mostly derived from the Euclidean domain. Such information is highly variable across brain geometries and severely hinders the training of modern machine learning algorithms. Geometric deep learning (Bronstein *et al.*, 2017) recently proposes to use convolutional filters on irregular graphs. However, geometric deep learning methods either lack the capability to process multiple surface domains or have spatial representations of surface data defined in a Euclidean space, ignoring the surface's complex geometry. Hence, there is a need for this thesis to develop a geometric framework that can directly learn and analyze cortical surface data across the population.

0.4 Research Objectives and Contributions

As highlighted in the challenges and limitations of the above subsection, the main objective of this research thesis is to **develop geometry aware tools that can aid in direct analysis of brain surface data**. We tackle this main objective by elaborating on four specific objectives. The first objective is to build a framework based on spectral graph theory for learning surface data via graph convolutions in neuroimaging. This novel approach enables a direct learning of surface data for a surface level segmentation. The second objective proposes to develop a learnable graph pooling for brain surfaces to perform subject level classification and regression tasks. The third objective presents a domain adaptation strategy to learn multiple surface data that is independent from any spectral domain with an application to cortical parcellation. The fourth objective is to build a method for the joint extraction and segmentation of cortical surfaces directly from MRI volume. These specific objectives of this thesis can be further detailed as follows:

1. **Spectral graph convolutions for cortical surface parcellation:** The first objective of this thesis is to learn surface data directly across multiple brain surfaces to separate the cerebral cortex into distinct regions based on structural information known as parcellation. Current conventional approaches (Fischl *et al.*, 2004b) typically rely on geometrical simplifications, such as spherical inflation and slow mesh deformations algorithms, often neglecting geometry and shape information. Graph-based deep learning (Bronstein *et al.*, 2017; Kipf & Welling, 2017; Monti, Boscaini, Masci, Rodolà, Svoboda & Bronstein, 2017) will be used to preserve geometry. However, geometric deep learning methods (Bronstein *et al.*, 2017; Kipf & Welling, 2017; Monti *et al.*, 2017) either lack the capability to process multiple surface domains or have spatial representations of surface data defined in a Euclidean space, ignoring the surfaces complex geometry. The principles from spectral graph theory will be exploited to formulate a function mapping multiple brains to a shared space. This formulation will enable learning across multiple surfaces. This research contribution is the first attempt at intrinsically learning surface data via spectral graph convolutions in neuroimaging to the best of our knowledge. This novel approach enables direct learning of surface data across compatible surface bases by exploiting spectral filters over intrinsic representations of surface neighborhoods. The results from the experiments indicate higher

parcellation accuracy and at par with the well-established FreeSurfer algorithm while gaining a drastic speed improvement from hours to seconds.

2. **Learnable pooling in graph convolutional networks for brain surface analysis:** The next objective of the thesis is to analyze surface data across the dataset to have a subject-level diagnosis, such as AD classification or birth age regression. A graph pooling operation is required to perform classification or regression directly on surface data represented as a graph. In the conventional Euclidean domain, pooling operation is straightforward with a fixed number of nodes and neighbors. However, a learnable pooling technique with geometry preserving features is essential for a brain surface with varying connectivity and size across the dataset. Contributions will be made by developing a learning framework to pool brain surfaces of arbitrary size hierarchically in spectral-domain for classification and regression tasks. The adaptive pooling technique will learn to aggregate information hierarchically depending on the output task. Various experiments will help to understand the effect of different parameters of the method. The learned regions for pooling and the performance of the method could potentially be used to reveal new biomarkers in the early diagnosis of diseases like Alzheimer's.

3. **Alignment invariant brain surface analysis:** The third objective of the thesis is to directly learn the surface data in a shared domain. Previous graph learning methods in the spectral domain depend on a mapping function (often referred to as spectral alignment) that projects the brain surfaces to shared space. This mapping enables learning of cortical data directly across multiple brain surfaces via graph convolutions. However, current spectral graph learning algorithms fail when brain surface data are misaligned across subjects, thereby requiring to apply a costly alignment procedure in pre-processing. Here, we focus on generalizing parcellation across multiple brain surface domains by removing the dependency on these domain-specific alignments. The graph domain adaptation method in an adversarial manner, learn a fully-convolutional GCN segmentator and a GCN domain discriminator,

both of which operate on the spectral components of surface graphs. The learned spectral filters of the GCN surface parcellation network is robust to differences in domain-specific alignment. Moreover, it also provides a better generalization for surfaces aligned to different references, without requiring an explicit re-alignment or manual annotations of these surfaces. The results from the experiments will validate the domain generalizability of the method, further increasing the overall computation time.

4. **Direct joint cortical surface reconstruction and segmentation from MRI volumes:** Cortical surface extraction from MRI volumes using traditional methods is computationally expensive (Fischl *et al.*, 1999). Recent advancements in deep learning has resulted in faster extraction of cortical surfaces (Cruz, Lebrat, Bourgeat, Fookes, Fripp & Salvado, 2021; Henschel *et al.*, 2020). However, surface analysis methods (Fischl *et al.*, 2004a; Lombaert, Criminisi & Ayache, 2015b) rely on the pre-reconstructed cortical surfaces. The final objective of the thesis is to jointly reconstruct and segment the brain surface directly from an input MRI volume. Existing traditional methods for these tasks follow a heuristic multi-step process that typically involves: the segmentation of brain tissues, the explicit computation of a surface mesh from segmentation masks, the geometric simplification of the mesh using non-linear projection techniques such as spherical inflation, the registration of the projected surface to an atlas, and finally the transfer of atlas labels to the registered mesh. These commonly-used tools for cortical reconstruction and parcellation, such as FreeSurfer, require extensive computation times, often in hours. A contribution to this topic will be made. The proposed method will learn a distance function to reconstruct white matter or pial surface and its corresponding spherical representation in the registered atlas space. The continuous representation of the spherical coordinates will extract an implicit iso-level brain surface and propagate the parcel labels from the parcellation atlas. The reconstruction and segmentation results on multiple datasets will highlight the robustness of the method with a drastic speed-up from hours to seconds of computation compared to FreeSurfer.

0.5 Thesis Outline

The organization of the work reported in the thesis is described in this section. This introductory chapter provided an overview and the context, motivation, and contribution of this thesis. **Chapter 1** presents the literature on the current state-of-the-art methods for parcellation, followed by introducing deep learning based method for surfaces (graphs) analysis and other useful topics relevant to this thesis. **Chapter 2** introduces the learning framework via spectral graph convolution for cortical surface parcellation. The work in this chapter is published in the Journal of Medical Image Analysis, titled *"Graph convolutions on spectral embeddings for cortical surface parcellation"*. **Chapter 3** discusses the adaptive pooling strategy proposed for surface classification tasks. This chapter corresponds to the paper entitled *"Learnable Pooling in Graph Convolution Networks for Brain Surface Analysis"* published in the IEEE Transactions on Pattern Analysis and Machine Intelligence. **Chapter 4** proposes a domain adaptation strategy for surface parcellation without any explicit alignment of graph eigenbases. This method allows spectral graph convolution network to work independent of the mapping function. The work presented in this chapter corresponds to the runner-up paper published in Graphs in Biomedical Image Analysis Second International MICCAI Workshop titled *"Graph Domain Adaptation for Alignment-Invariant Brain Surface Segmentation"*. **Chapter 5** presents the final contribution of the thesis describing the joint reconstruction and segmentation of brain surface directly from input MRI volume. The content of this chapter corresponds to the paper *"SegRecon: Learning Joint Brain Surface Reconstruction and Segmentation from Images"* submitted to the Journal of Medical Image Analysis. The **Conclusion Chapter** summarizes the contributions of this thesis and discusses its limitations and possible future extensions. Finally, **Appendix I** provides a novel approach for learning the transformation matrix required for aligning brain meshes in a direct data-driven way and **Appendix II** provides link to the codes.

0.6 Guidelines for reading the thesis

This section presents suggestions for the readers studying this thesis to find essential information. As described before, the overall goal of this thesis is to develop novel geometry-aware approaches for brain surface analysis. This thesis may, therefore, be read in multiple ways to obtain specific information.

Cortical surface parcellation – The reader interested in surface parcellation method may focus in Chapter 2, 4 and 5. Specifically, Sec. 2.2 performs parcellation using spectral graph convolution in the first approach, given the cortical mesh. Secondly, given the cortical mesh, the readers can look to Sec. 4.2.2 to perform domain invariant surface analysis with application to cortical parcellation. Finally, the readers may focus particularly on Sec. 5.2 for cortical parcellation directly from the MRI volume space and extract cortical surfaces. Sec. A.I.3.2 provides a data-driven methodology to learn the alignment matrix for multiple brain surfaces, with application to cortical parcellation.

Surface classification or regression – The reader interested in performing surface based classification or regression tasks across the population may focus on Sec. 3.2.3. Here, a learning pooling strategy to down sample the original surface is used for disease classification or age regression. The readers can use this framework to obtain a lower-dimensional representation of the brain surface for other downstream tasks.

Cortical surface reconstruction – The reader interested in tools for cortical surface reconstruction and obtaining atlas labels on the surface may focus on Sec. 5.2. The first part predicts a signed distance function for the reconstructing surface, and the second part uses the implicit surface representation to reconstruct a cortical mesh.

General surface analysis – The reader interested in the general applicability of the developed surface analysis may focus on Sec. 2.2 and Sec. 3.2.3. The first performs geometry-aware node classification of graphs with the potential to regressing features on the surface. Sec 3.2.3 provides an adaptive graph pooling technique useful for lower dimensional representation or encoder-decoder networks. The last Sec. 5.2 may be applicable for extracting surfaces or point cloud data from volumetric space by labeling the surfaces.

0.7 Published Work

The findings in this thesis have led to the following publications.

– Journals:

1. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. “Learning Joint Surface Reconstruction and Segmentation, from Brain Images to Cortical Surface Parcellation”. *Medical Image Analysis - 2021. Under review*

2. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. “Learnable Pooling in Graph Convolution Networks for Brain Surface Analysis”. *IEEE Transactions on Pattern Analysis and Machine Intelligence - 2020*

3. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. “Graph convolutions on spectral embeddings for cortical surface parcellation”. *Medical Image Analysis - 2018*

4. Dolz Jose, **Gopinath Karthik**, Yuan Jing, Lombaert Herve, Desrosiers Christian,, Ben Ayed Ismail. “HyperDense-Net: A hyper-densely connected CNN for multi-modal image segmentation”. *IEEE Transactions on Medical Imaging - 2018*

– Conferences:

1. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "SegRecon: Learning Joint Brain Surface Reconstruction and Segmentation from Images". *International Conference on Medical Image Computing and Computer Assisted Intervention - 2021.*

2. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "Graph Domain Adaptation for Alignment-Invariant Brain Surface Segmentation". *International Conference on Medical Image Computing and Computer Assisted Intervention – GRAIL - 2020 [**Runner up**].*

3. **Gopinath Karthik**[1], Ran He[1], Desrosiers Christian, Lombaert Herve. "Spectral Graph Transformer Networks for Brain Surface Parcellations". *IEEE International Symposium on Biomedical Imaging - 2019.*

4. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "Adaptive Graph Convolution Pooling for Brain Surface Analysis". *International Conference on Information Processing in Medical Imaging - 2019.*

5. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "Graph Convolutions on Spectral Embeddings: Learning of Cortical Surface Data". *Medical Imaging meets NeuirIPS - 2018.*

– Short papers:

1. **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "Cortical Parcellation via Spectral Graph Convolutions". *International Conference on Medical Imaging with Deep Learning - 2019.*

2. Reddy Charan, **Gopinath Karthik**, Desrosiers Christian, Lombaert Herve. "Brain tumor segmentation using topological loss in convolutional networks". *International Conference on Medical Imaging with Deep Learning - 2019.*

---

[1] equal contribution

CHAPTER 1

BACKGROUND

1.1 Overview

The human cerebral cortex is a complex structure, the understanding of which requires a division into anatomically-relevant cortical areas. As a case study, we did a literature survey of review papers focusing on parcellation of the cortical surface. In the field of neuroscience, obtaining an accurate marking is a problem that has been studied for a few decades. Fig. 1.1 shows the representation of the brain and its primary cortical areas. There are roughly around 30 (Van Essen, Glasser, Dierker, Harwell & Coalson, 2011) to 200 regions (Nieuwenhuys, 2013) per hemisphere. Few works (Fischl & Dale, 2000; Nieuwenhuys, 2013) attempt to estimate surface statistics such as the volume of a region or the thickness of the brain surface with fully automatic or semi-automatic methods. We provide a brief overview of standard structural parcellation methods below.

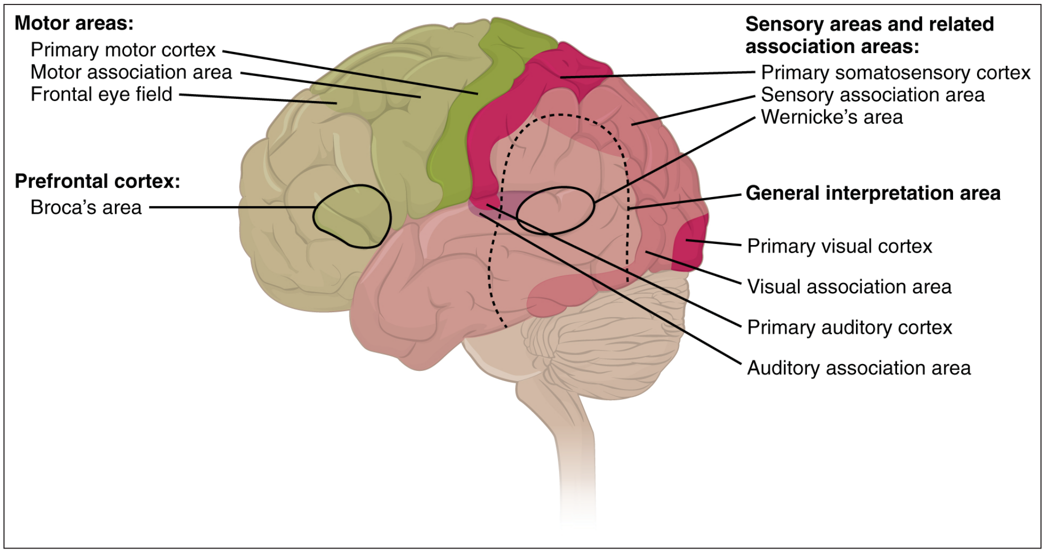


Figure 1.1 Primal regions of brain – A human brain divided into primal regions. Taken from Anatomy and Physiology (2013)

1.2 Surface Analysis Methods

Human brain parcellation is one of the necessary steps to obtain quantitative measurements of anatomical regions. The most expensive way is to manually map the regions by delineating the region boundary for numerous subjects. Fig. 1.2 illustrates parcellation of the brain into cortical segments. Many researchers have proposed algorithms to automate this process. Manual segmentation is necessary for any supervised algorithm to learn the region boundary. Previous parcellation techniques (Fischl *et al.*, 2004b) use a blurry average template to obtain brain areas across subjects.

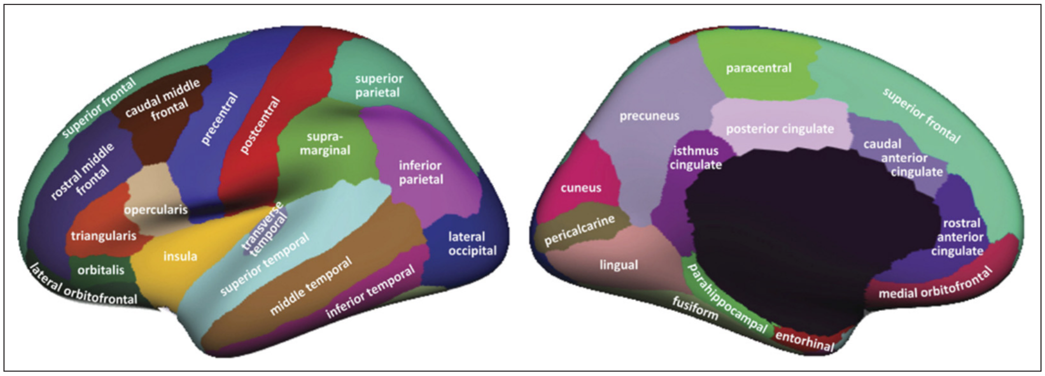


Figure 1.2 Cortical parcellation of a brain. Taken from Klein & Tourville (2012, p. 3)

1.2.1 FreeSurfer

FreeSurfer is an open-source software for neuroimage processing, which includes a comprehensive set of tools for cortical surface analysis. This software suite is widely used withing the neuroimaging community to study the human brain. The standard processing pipeline for MRI consists of various steps (Dale *et al.*, 1999; Fischl *et al.*, 1999), presented next.

The initial step is to align each volume to an atlas with affine registration. Next, the FreeSurfer performs intensity standardization for every subject by measuring the bias field in the white matter intensity. The skull region is not necessary for processing, and hence the skull is stripped

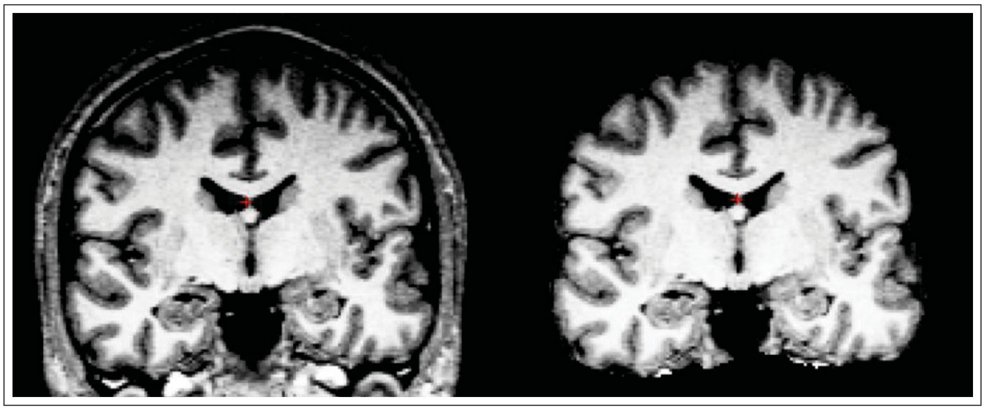
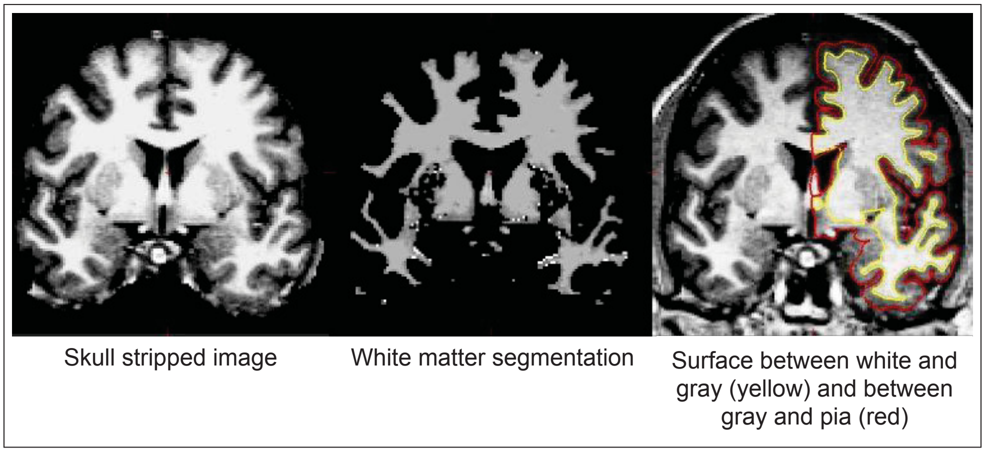


Figure 1.3 FreeSurfer stages – MRI scans with skull stripped brain. Taken from Dale *et al.* (1999, p. 184)



Skull stripped image

White matter segmentation

Surface between white and gray (yellow) and between gray and pia (red)

Figure 1.4 FreeSurfer surface extraction – FreeSurfer stages for obtaining cortical surface. Adapted from Dale *et al.* (1999, p. 186)

using a deformable template model (Ségonne, Dale, Busa, Glessner, Salat, Hahn & Fischl, 2004). Fig. 1.3 shows the original MRI scan (left) and the skull stripped brain (right). Each voxel is then classified as background or white matter based on intensity and neighborhood information. Next, a cutting plane separates the brain into two hemispheres, then the cerebellum and the brain stem connecting the two hemispheres are removed. The intensity gradient between the gray matter and CSF results in a surface called the pial surface. Fig. 1.4 shows these initial steps of the FreeSurfer pipeline. Cortical thickness is measured at each location by considering the distance between the white and the pial surface (Fischl & Dale, 2000). Fig. 1.5 shows a 3D view of the pial surface along with its inflated version.

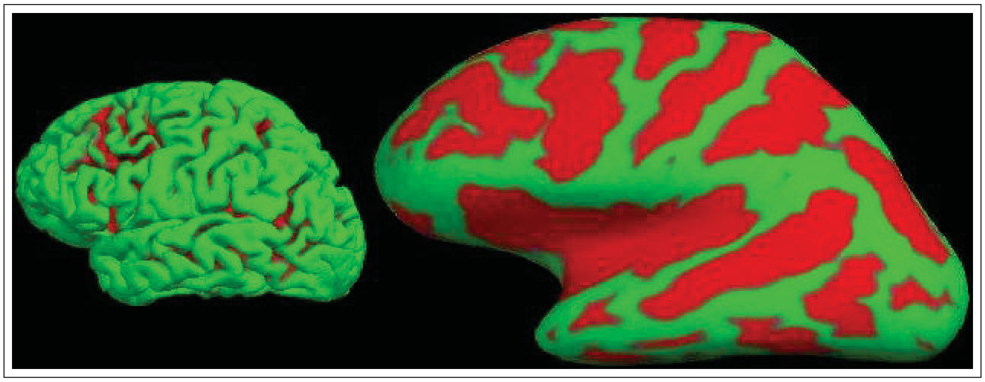


Figure 1.5 Visualization of surfaces – 3D FreeSurfer pial (left) and inflated (right) surface. Taken from Fischl *et al.* (1999, p. 202)

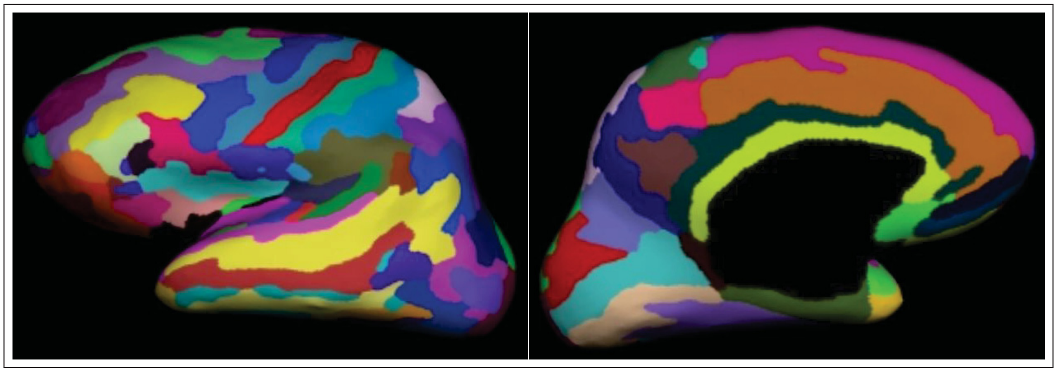


Figure 1.6 Cortical regions – Surface-based labeling from FreeSurfer. Taken from Fischl *et al.* (2004b, p. 12)

Once the surface is obtained, it is mapped to a standard MNI305 spherical space. FreeSurfer has an atlas produced from the training set with manual labels, which are also propagated onto the spherical space. A pointwise correspondence for a subject is achieved to match with the atlas. For each brain, the vertex on the surface or voxels of the volumes is a point. The curvature in each of the principal directions at a vertex forms the information for surfaced-based labeling. The segmentation maximizes the probability of the input point belonging to the particular labeled class. The class which has the highest probability gets the label assigned, thus creating an initial segmentation. Fig. 1.6 shows parcellation on a 3D surface obtained from the FreeSurfer pipeline.

1.3 Convolutional Neural Network based Methods

Deep learning is a class of machine learning methods that uses neural network architecture for learning a hierchical feature representation is a data-driven manner. Convolutional neural networks (CNNs) are a widely-used method for solving various computer vision tasks. Developments in semantic segmentation have allowed their use to medical image segmentation. CNNs typically use two-dimensional filters for computer vision applications. However, for medical images segmentation, recent publications have shown the advantage of considering volumetric context with three-dimensional kernels.

In neuroimaging, CNNs are commonly used for the segmentation of cortical surfaces, subcortical structures, or lesions, as well as for disease prediction (Havaei, Davy, Warde-Farley, Biard, Courville, Bengio, Pal, Jodoin & Larochelle, 2017; Shen, Wu & Suk, 2017) (Valverde, Cabezas, Roura, González-Villà, Pareto, Vilanova, Ramió-Torrentà, Rovira, Oliver & Lladó, 2017). A widely used method for image segmentation is to classify individual pixels based on small image patches (both 2-dimensional and 3-dimensional) extracted around these pixels. This technique has been used for segmentation tasks in MRI images, for example in brain tumor segmentation (Havaei *et al.*, 2017), white matter segmentation in multiple sclerosis patients (Valverde *et al.*, 2017) and segmentation of subcortical structures (Wachinger, Reuter & Klein, 2017).

The first application of CNNs to medical image segmentation was on electron microscopy images (Ciresan, Giusti, Gambardella & Schmidhuber, 2012). In that study, segmentation was performed by using 2D convolution stacks. The approach aimed at classifying individual pixel of a slice by extracting a neighborhood patch. However, applying a sliding window to obtain patched is time-consuming and results in the redundant computation for overlapping regions. The other issue was with the inability of the network to learn global features. To overcome

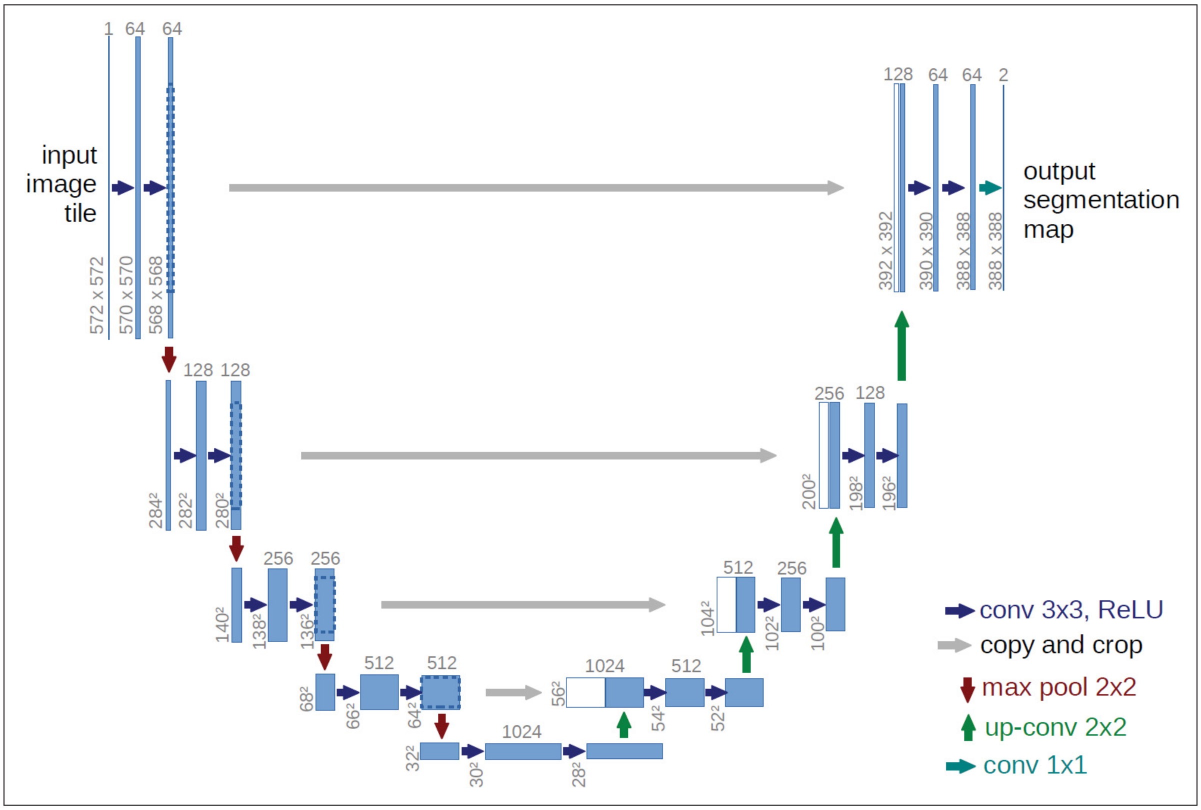


Figure 1.7 Overview of U-net architecture for medical image segmentation. Taken from Ronneberger *et al.* (2015, p. 235)

these limitations, Ronneberger et al. proposed a fully-convolutional approach (Ronneberger *et al.*, 2015). The fully-convolutional network (FCN) produces output the same size of the input making it independent of the image size. Fig. 1.7 show an example of such architecture. The design is similar to that of convolutional encoder-decoders consisting of contracting and expanding stage. During the contraction step, the size of the input gets reduced while the number of feature maps increases. The reverse happens in the expanding phase. The final stage uses a convolution filter of size 1 × 1 producing the segmentation map. The 'U' shaped appearance of the architecture results in the name U-net. A slice by slice extension of this work with 3D convolutional kernels for 3D application also exists (Çiçek, Abdulkadir, Lienkamp, Brox & Ronneberger, 2016; Milletari, Navab & Ahmadi, 2016). The varying size of structures and organs makes slice-by-slice implementations of a 2D network inefficient. The incomplete labeled data in these cases were handled by applying transformations like shifting, rotating and scaling images, as well as augmenting grey values. However, this random deformation of the original image at the beginning of the training may result in training on a different version of the original dataset.

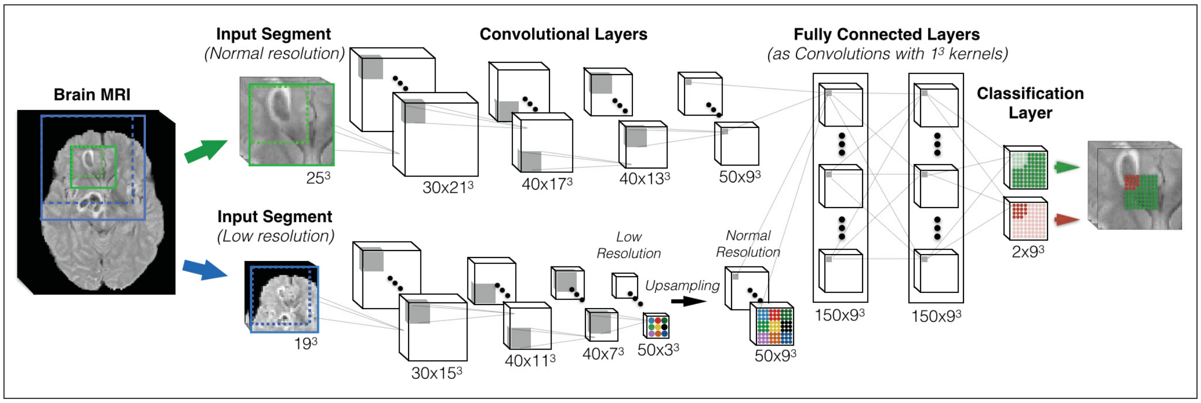


Figure 1.8 Overview of DeepMedic architecture for MRI image segmentation. Taken from Kamnitsas *et al.* (2017b, p. 66)

Other FCN-based medical image segmentation methods have been reported in the literature (Chen, Qi, Yu & Heng, 2016; Chen, Dou, Yu, Qin & Heng, 2018a; Dou, Chen, Jin, Yu, Qin & Heng, 2016). These methods are extended versions of U-net, which omit the expanding path in this architecture, in combination with the low-resolution segmentation map deconvolved to the original resolution. These low-resolution outputs are combined with a fully convolutional network to produce the final segmentation. Another characteristic of these works is the use of deep supervision. Here, feature maps from earlier points in the network are used to create secondary segmentation maps. DeepMedic (Kamnitsas *et al.*, 2017b) is another popular 3D CNN architecture, which has been used for various segmentation tasks involving brain MR images. Fig. 1.8 depicts this architecture. One of the main contributions of this architecture is that it consists of two paths, the first one receiving the subregion of the original image that is to be segmented, and the second receiving a larger region downsampled to a lower resolution before being fed to the network. This architecture enables the network to learn global image features. Likewise, (Witten, Frank, Hall & Pal, 2016) followed a similar approach, using 2D convolution on 2D slices instead of 3D convolutions. As postprocessing step, a conditional random field (CRF) is applied to the output to get smoother regions.

1.3.1 FastSurfer

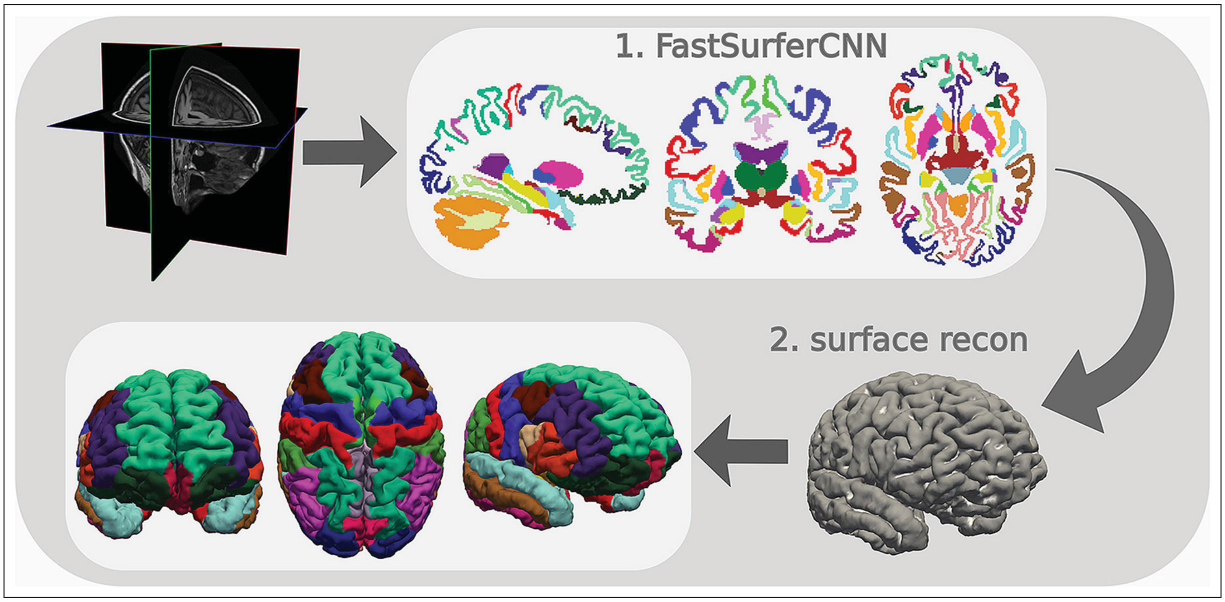


Figure 1.9 Overview of FastSurfer pipeline for cortical surface extraction. Taken from Henschel *et al.* (2020, p. 3)

Fastsurfer (Henschel *et al.*, 2020) similar to FreeSurfer is a neuroimaging tool for cortical surface analysis. This software used deep learning based tools to speed up the FreeSurfer based pipeline. FastSurfer uses deep learning architecture called fastsurfer CNN to create a whole brain segmentation into 95 classes in less than a minute. The CNN architecture used in this pipeline is composed of three fully convolutional neural networks operating on coronal, axial and sagittal 2D slices with a final aggregation stage. Each F-CNN incorporates both local and global competitive dense blocks and competitive skip pathways to accurately segment cortical and subcortical structures. In addition to performance and computation gains, the use of CNN for FastSurfer avoids using traditional FreeSurfer steps such as skull tripping and nonlinear registration. Fig. 1.9 shows the overall pipeline of the FastSurfer algorithm.

For surface extraction and analysis, FastSurfer introduces alternatives to FreeSurfer's traditional iterative spherical inflation process. The surface is first mapped to the spectral embedding using the first three non constant eigenfunctions of the Laplace-Beltrami operator and then the 3D spectral embedding vector is scaled to unit length to obtain the final spherical map. This process of spherical mapping is slightly faster computationally compared to FreeSurfer. The FastSurfer pipeline is validated on multiple unseen datasets with varying age ranges, diseases states and acquisition protocols to ensure generalizability and robustness.

1.3.2 Cortical Surface Extraction - DeepCSR

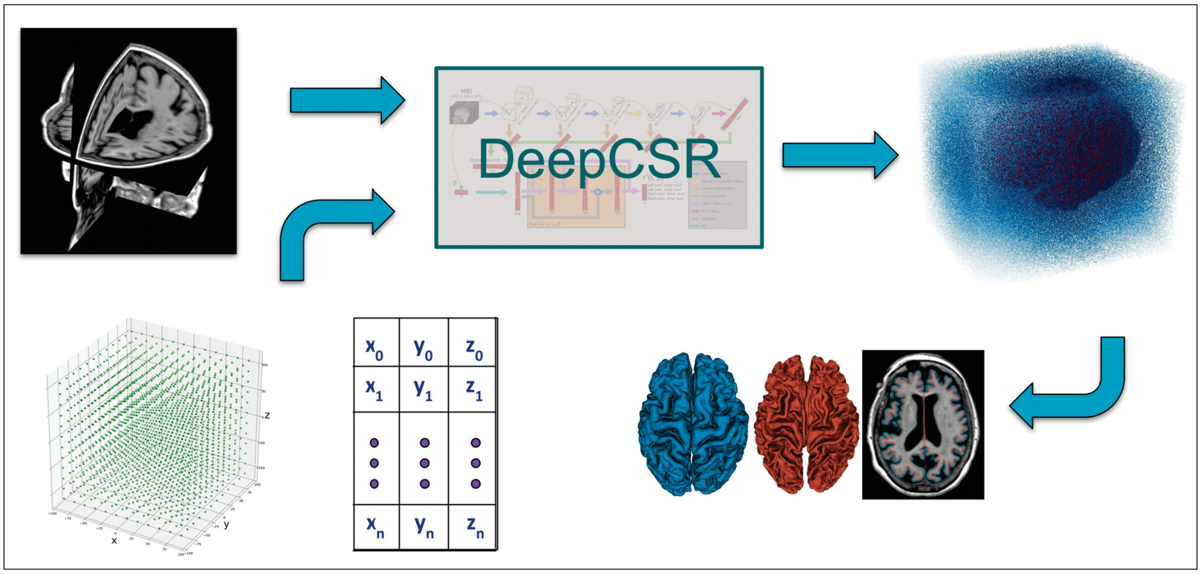


Figure 1.10 Overview of DeepCSR algorithm for surface extraction. Taken from Cruz (2021)

Reconstructing the cortical surfaces from brain MRI is essential to a wide range of brain analyses (Fischl *et al.*, 1999; Tosun, Rettmann, Naiman, Resnick, Kraut & Prince, 2006). However, most approaches follows a multi-step slow process. For instance, FreeSurfer (Fischl *et al.*, 1999) follows 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. With advancements in deep learning, a faster algorithm for automatically extracting the brain surface was proposed, namely FastSurfer (Henschel *et al.*, 2020). This proposed method performs deep learning based tissue segmentation and follows sequential steps similar to FreeSurfer (Fischl *et al.*, 1999) for surface extraction.

In computer vision, several deep learning based methods (Guo, Wang, Hu, Liu, Liu & Bennamoun, 2020; Park, Florence, Straub, Newcombe & Lovegrove, 2019; Qi, Yi, Su & Guibas, 2017b) have been proposed to extract point clouds, graphs, or surfaces from images and volumes. Park *et al.* (2019) proposed a deep learning based model called DeepSDF that learns a continuous Signed Distance Function (SDF) from 3D volume to represent the shape. This continuous volumetric representation implicitly encodes the boundary of the shape as a zero-level-set function without the need for an explicit segmentation step. Inspired by this, Cruz *et al.* (2021) proposed a model, called DeepCSR, for the reconstruction of cortical surfaces. The advantages of having multiple points for sampling from a reference grid allow DeepCSR to reconstruct cortical surfaces at a different resolution. Fig. 1.10 shows the architecture of the DeepCSR network for cortical surface reconstruction. However, these processes are costly in both computation and memory for detailed surfaces with hundreds of thousands of points. Compared to the traditional FreeSurfer, these surface reconstruction methods are also limited only to surface extraction.

To conclude, the variety of CNN-based medical image segmentation methods is mainly due to different attempts at addressing difficulties specific to medical images. For our work, we aim to develop a learning algorithm that works directly on the surface of the brain, and not in image space. The surface of the brain is geometry-driven and contains many folds. The distance between two points on the surface is geodesic, and Euclidean distance may not be suitable to model surface proximity. These limitations impede the use of standard CNN techniques for learning on directly surfaces. However, recent advancements in geometric deep learning provide an powerful way to overcome these limitations.

1.4 Geometric Deep Learning

Methods trying to generalize deep neural networks to non-Euclidean domains such as graphs and manifolds fall under the term *geometric deep learning*. Neural network approaches are popular and widely used for data represented in the Euclidean space. Attempts have previously been made to develop algorithms to work on data lying on non-Euclidean space. Geometric data, such as social networks in computational social sciences and surfaces meshes in brain imaging, are complex for standard machine learning techniques.

Bronstein *et al.* (2017) review various deep neural methods to learn on graphs and manifold. The survey (Bronstein *et al.*, 2017) provides an overview of conventional deep learning techniques explaining the crucial assumptions of convolutional network architectures and the underlying data. The notion of graph theory with the differences and similarities between Euclidean and non-Euclidean learning methods are explained in this work. This work (Bronstein *et al.*, 2017) provides fundamentals to several medical applications where data inherently lie on surfaces or graphs. Fig. 1.11 shows an example of the different type of graphs in neuroimaging. Next, we explain the basics of graphs and the graph theory behind the geometric deep learning methods. A graph is a mathematical object mostly used to model the relation between two set of points.

A graph is defined mathematically as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$, with $\mathcal{V}$ being the set of nodes or vertices $|V| = n$. $\mathcal{E}$ represents set of edges with $e_{ij} \in \mathcal{E}$ and $\mathcal{W}$ is the corresponding adjacency matrix. The principles from spectral graph theory are used to analyze the data lying on the graph. The Laplacian matrix L, defined as the difference between the degree matrix D and adjacency matrix W, i.e., $L = D - W$, is a powerful operator often used to process the graph networks. The decomposition of the graph Laplacian $L = U \Lambda U^T$ produces the standard Fourier bases of the signal that can be used for graph Fourier transforms.

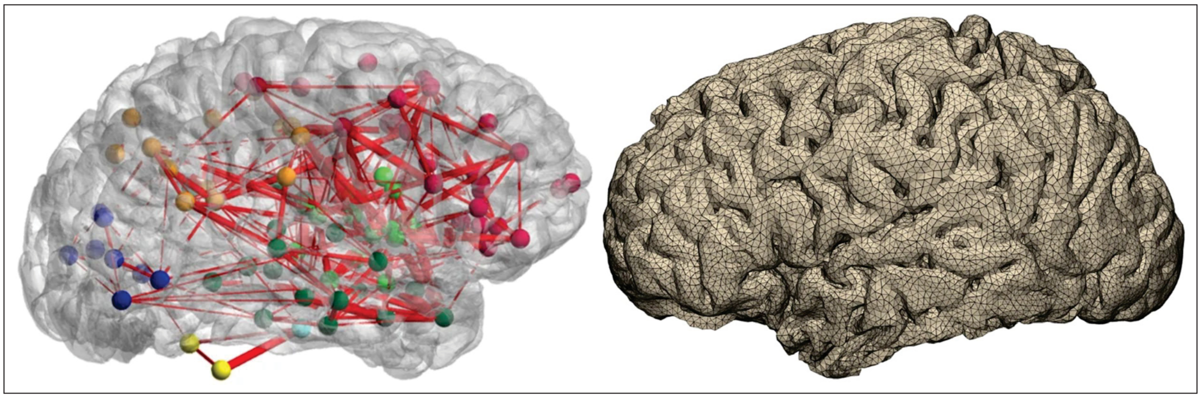


Figure 1.11 Different graph structures – Left: indicates the brain structural connectivity network. Adapted from Chu *et al.* (2018, p. 2). Right: indicates the constructed 3D brain mesh. Adapted from Javan *et al.* (2020, p. 327)

Deriving the analogies from the conventional CNNs, there are two popular (spectral or spatial) approaches for graph convolution network.

1.4.1 Spectral Methods

Spectral methods focuses on mapping data such as graphs or manifolds into the spectral domain using the spectrum (eigenvalues and/or eigenfunctions). We first explain spectral shape analysis in neuroimaging followed by convolution networks in spectral domain.

1.4.1.1 Spectral Shape Analysis

Statistical analysis of geometric shapes involves studying varying properties of shapes such as curvature, geodesic distance and angles. In particular, the spectrum based methods are inspired from signal processing, where, the input signal is mapped to the Fourier bases using the Fourier transform. The eigenvalues of the Laplace-Beltrami operator on a manifold defines the shape spectrum (Reuter, Wolter & Peinecke, 2006). For discrete meshes or graphs, the spectral decomposition of the graph Laplacian matrix defined on vertices and edge connections provides a spectrum descriptor (Lombaert, Grady, Polimeni & Cheriet, 2012). Fig. 1.12 shows the spectrum for two brain surfaces. This shape descriptor is widely used in literature for shape recognition, matching, and registration.

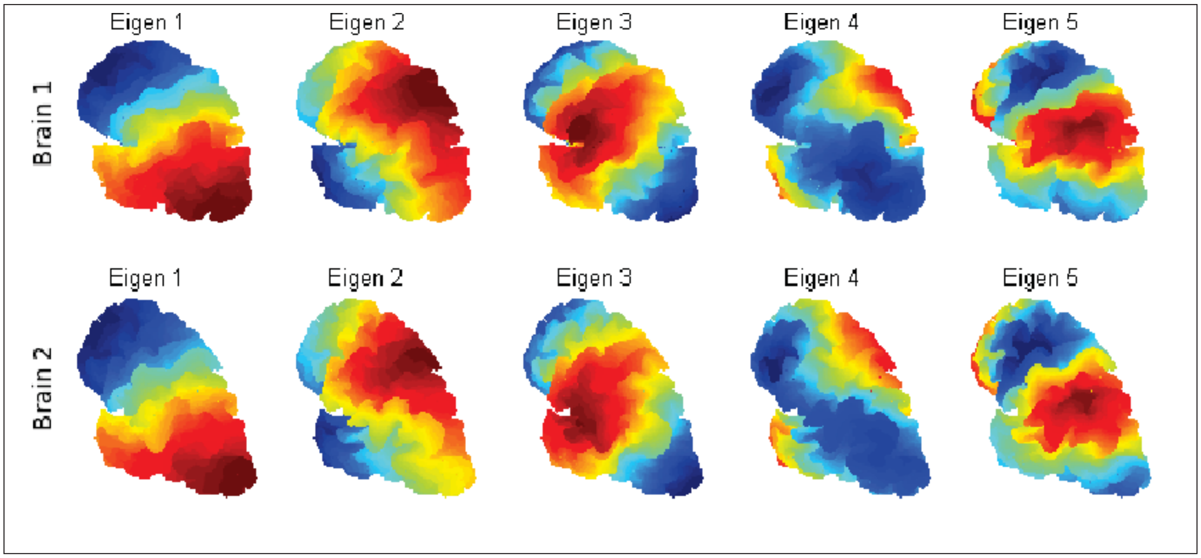


Figure 1.12 Eigen decomposition of graph Laplacian – The first six eigenvectors of two brain surfaces. Taken from Lombaert *et al.* (2011, p. 661)

Reuter *et al.* (2006) proposed ShapeDNA, the first global spectral shape descriptors using the eigenvalues of the Laplace–Beltrami operator for retrieval of shapes. The Laplace-Beltrami operator has also been used for medical shape analysis with no to minimal shape pre-processing (Niethammer, Reuter, Wolter, Bouix, Peinecke, Koo & Shenton, 2007). In neuroimaging, Wachinger, Golland, Kremen, Fischl, Reuter, Initiative et al. (2015) proposed BrainPrint using the same Laplace-Beltrami operator to produce discriminative representation of brain morphology. This descriptor allows effective comparison of brains and proved efficient for multiple tasks including subject identification, age and sex prediction and brain asymmetry analysis. Reuter (2010) used the spectrum of Laplace-Beltrami to find geometric patterns of the cortical folding.

Finding correspondence between shapes and objects has also been a challenging and important task for shape analysis. Shapiro & Brady (1992); Spielman (2012) were the first to propose a spectral method to find feature based correspondence. Depending on the proximity matrix, different shape correspondence methods were proposed using the spectrum. Specifically, Bronstein, Bronstein & Kimmel (2006,0) proposed the use of Multi-Dimensional Scaling matrix for correspondence problem. For images, Lombaert, Grady, Pennec, Ayache & Cheriet (2014) proposed Spectral Log-Demons for image registration using spectral eigenvalues that captures global shape characteristics. Eigendecomposition of heat kernels called heat signature kernels is used as local descriptors by Sun, Ovsjanikov & Guibas (2009) to perform for detecting shape symmetry or partial matching. Similarly, Laplacian eigenfunctions are used in medical image analysis problems such as point registration (Mateus, Horaud, Knossow, Cuzzolin & Boyer, 2008), diffeomorphic matching of cortical surfaces (Lombaert, Sporring & Siddiqi, 2013), and shape registration (Sharma, Horaud & Mateus, 2021).

1.4.1.2 Spectral Graph Convolutions

The convolution theorem on a graph is similar to defining the convolution operation in the spectral domain. Recently, graph convolutions based on spectral graph theory have been proposed to work on graphs and manifolds (Bruna, Zaremba, Szlam & Lecun, 2014b; Henaff, Bruna & LeCun, 2015). An approximate version of smooth filters was designed in the spectral domain using Chebyshev polynomials and trained as a neural network model (Shuman, Vandergheynst, Kressner & Frossard, 2011). In (Kipf & Welling, 2017), Kipf & Welling (2017) proposed a similar spectral graph convolution approach, which was shown to have a faster training and higher accuracy on many benchmark graph datasets. This approach is illustrated in Fig. 1.13.

The work in (Chen, Zhu & Song, 2018b) overcomes the memory issues faced by (Kipf & Welling, 2017) for large graphs by proposing a sampling-based stochastic algorithm. Yet, a common limitation of these spectral methods is that the Fourier bases vary across different graphs. Thus, these methods are not directly applicable to multiple graph learning.

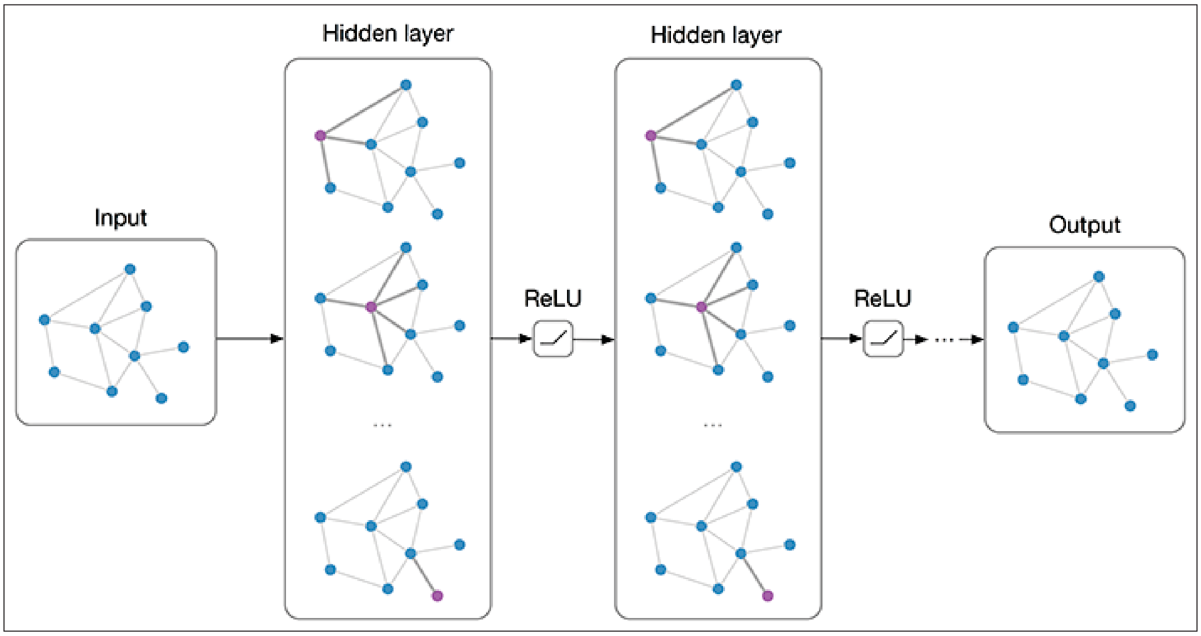


Figure 1.13 Overview of graph convolution network – An example of Spectral graph convolution network for node classification. Taken from Kipf (2016)

1.4.2 Spatial Methods

Spatial approaches define convolution as standard template matching following conventional spatial domain techniques. The fundamental challenge with these methods is identifying the ordering of graphs and its neighborhood. A spatial graph convolution approach proposed by (Atwood & Towsley, 2016) considers a random walk diffusion process on graph. The relation between any two nodes is captured by a weight function by $w_i\left(x, x'\right) = e^{-\left(d(x,x')-\rho_i\right)^2/2\sigma^2}$. In (Kokkinos *et al.*, 2012), CNNs are adapted to geodesic polar coordinates by constructing a patch operator. The weighting function of the patch operator localized around ρ, θ is given by $w_{ij}\left(x, x'\right) = e^{-\left(\rho(x')-\rho_i\right)^2/2\sigma_\rho^2} \cdot e^{-\left(\theta(x')-\theta_j\right)^2/2\sigma_\theta^2}$. Fig. 1.14 shows the spatial convolution method, based on the diffusion of the geodesic polar coordinates filters, on a toy example.

Monti *et al.* (2017) proposed a generic graph convolution based on learnable patch operators in the spatial domain. This work aims at constructing a patch operator that maps the node i and its neighboring node $j \in \mathcal{N}(is)$ a to a D-dimensional pseudo-coordinates $u(i, j)$ and feeding it a

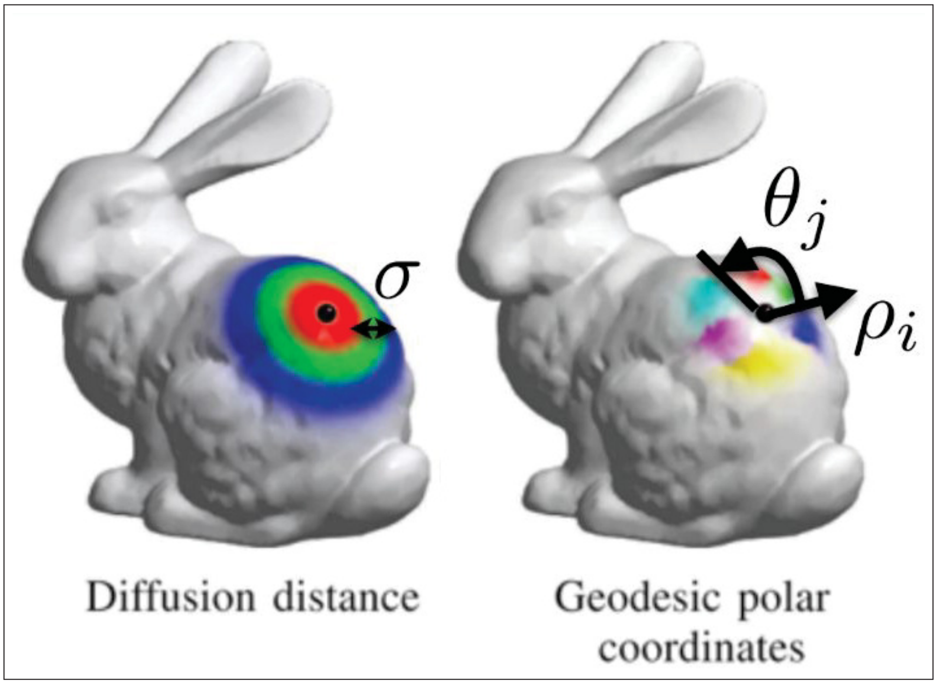


Figure 1.14 Illustration of Spatial graph convolution networks – Diffusion based filtering (Atwood & Towsley, 2016) and geodesic polar coordinate (Kokkinos *et al.*, 2012) based spatial graph convolution (Bronstein *et al.*, 2017). Adapted from Bronstein *et al.* (2017, p. 34)

learnable Gaussian kernel functions. The graph convolution in the spatial domain is then based on this patch operator. Fey et al. (Fey, Lenssen, Weichert & Müller, 2018b) follows the work in (Monti *et al.*, 2017), but instead use B-spline kernels for the graph convolution. As an alternative to the simple mean aggregator function to combine the signals from graph nodes, Hamilton et al. (Hamilton, Ying & Leskovec, 2017) propose to use LSTM aggregators and pooling aggregators.

In (Veličković, Cucurull, Casanova, Romero, Liò & Bengio, 2018), Velivckovic et al. present an attention-based model to weight the neighboring nodes by some learnable importance. The attention layer contains a shared learnable weight matrix with the attention coefficients between node i and its neighbor node $j \in \mathcal{N}(i)$ encoding the importance of feature vector of the neighboring node j for node i.

Spatial approaches overcome the limitations of spectral methods. However, these methods do not fully exploit the geometric information.

1.4.3 Geometric Deep Learning: Application to Neuroimaging

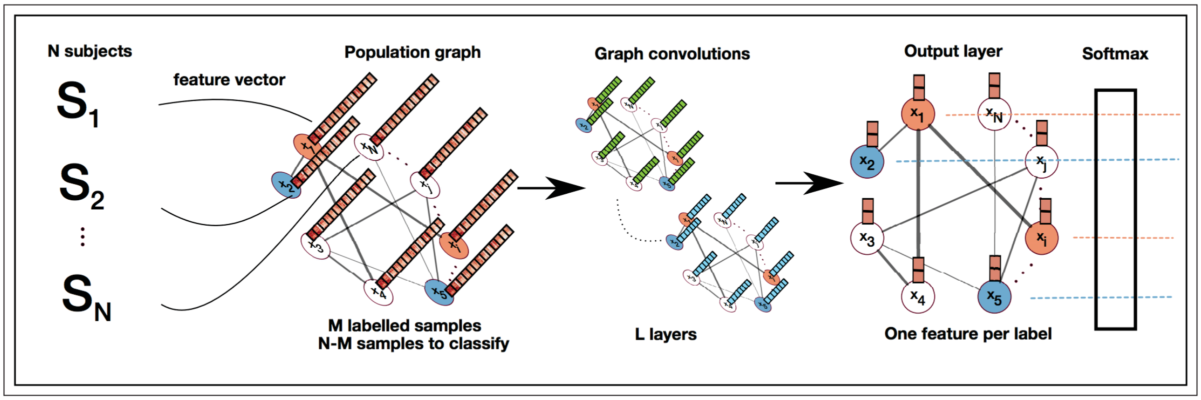


Figure 1.15 Overview of Spectral graph convolution for disease prediction. Taken from Parisot *et al.* (2017, p. 179)

Neuroimaging is an area where signals are naturally measured on a non-Euclidean domain. Geometric deep learning would thus be useful for developing tools to analyze these data. Recently, Parisot *et al.* (2017) used a graph CNN to detect autism from the functional brain networks. Fig. 1.15 shows the overview of their method. The goal of this work was to assign to each subject a label (e.g., control or diseased) describing the subject's disease. Toward this goal, the method constructs a population graph with each vertex represented by a patient with a feature vector. The graph edge weights are from phenotype data. The adjacency matrix of the graph defines the similarity between the imaging features across subjects. The experiment was carried out to perform diagnosis of autism across subjects. Another application of geometric deep learning to neuroimaging, closely related to the current research proposal, is the recent work of Cucurull, Wagstyl, Casanova, Veličković, Jakobsen, Drozdzal, Romero, Evans & Bengio (2018) on cortical mesh parcellation. This work used the attention-based model of (Veličković *et al.*, 2018) to segmentation two regions inside Broca's area.

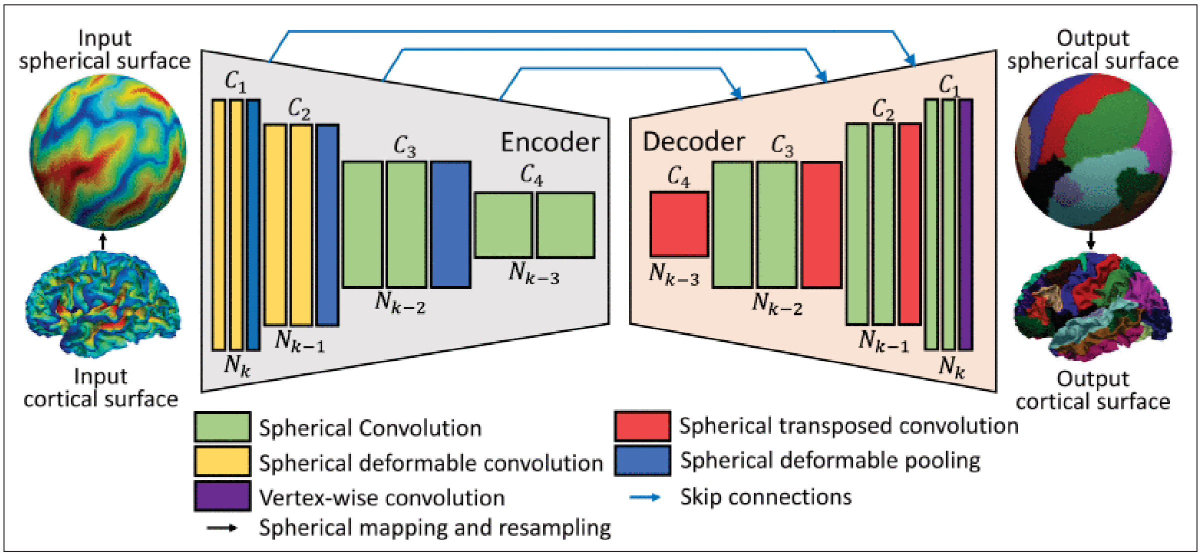


Figure 1.16 Overview of Spherical UNet proposed for cortical parcellation. Taken from Zhao *et al.* (2021, p. 1223)

Graph neural networks or graph convolution networks have also been applied to solve multiple other neuroimaging problems (Ahmedt-Aristizabal, Armin, Denman, Fookes & Petersson, 2021) and identifying diseases (Ktena, Parisot, Ferrante, Rajchl, Lee, Glocker & Rueckert, 2018). For instance, Rakhimberdina & Murata (2019) proposed analysis of functional connectivity fMRI data for disease classification. Methods such as (Kazi, Shekarforoush, Krishna, Burwinkel, Vivar, Kortüm, Ahmadi, Albarqouni & Navab, 2019; Kazi, Cosmo, Navab & Bronstein, 2020) perform node-level classification for population graphs to predict the disease class. Similarly to classification, GCNs have also been used for the segmentation of anatomical shapes from images (Selvan, Kipf, Welling, Juarez, Pedersen, Petersen & de Bruijne, 2020; Yan, Youyong, Jiasong, Coatrieux & Huazhong, 2019). A Spherical U-Net architecture was proposed (Zhao *et al.*, 2021) for cortical parcellation of the brain surface in the spherical space. Fig. 1.16 shows the architecture of the spherical UNet network for cortical parcellation. At volume level, Yan *et al.* (2019) proposed a GCN based segmentation model to classify supervoxels into different brain tissue types.

1.5 Domain Adaptation for Medical Image Analysis

One of the objective of this thesis is to learn alignment invariant brain surface parcellation with domain adaptation. In this section, we give a review of important and recent literature on this domain adaptation topic.

Convolutional neural networks provide state-of-the-art performance for most image analysis tasks, including image classification, registration, and segmentation(Arbabshirani *et al.*, 2017). However, to learn the data distribution, these models typically require large annotated datasets for training, which are often expensive to obtain in medical applications. Additionally, these deep learning models lack robustness to the distribution difference between the training and test data. To overcome these limitations, there are techniques in machine learning known as domain adaptation that allow model trained on source domain samples to generalize on samples from target domain. Fig. 1.17 shows an overview of an adversarial domain adaptation network for segmentation.

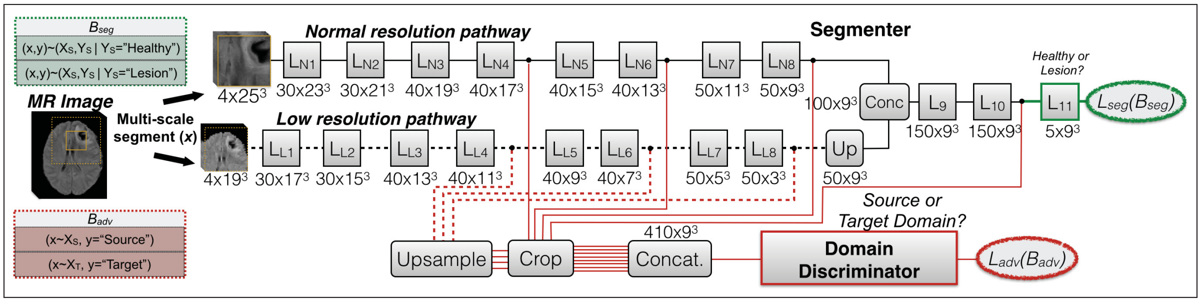


Figure 1.17 Overview of adversarial domain adaptation network for segmentation. Taken from Kamnitsas *et al.* (2017a, p. 600)

In computer vision, Generative adversarial networks (GANs) use adversarial training to produce realistic images (Goodfellow, Pouget-Abadie, Mirza, Xu, Warde-Farley, Ozair, Courville & Bengio, 2014). In medical imaging, GANs have been used in particular to explore the structures intrinsic to data (Yi, Walia & Babyn, 2019). Specifically, GANs have been explored for super-resolution image synthesis (Chen *et al.*, 2018a; Sánchez & Vilaplana, 2018), image segmentation (Delisle, Anctil-Robitaille, Desrosiers & Lombaert, 2021; Kamnitsas, Bai, Ferrante, McDonagh, Sinclair, Pawlowski, Rajchl, Lee, Kainz, Rueckert & Glocker, 2018; Yang, Xu, Zhou, Georgescu, Chen, Grbic, Metaxas & Comaniciu, 2017), classification (Madani, Moradi, Karargyris & Syeda-Mahmood, 2018), and image registration (Fan, Cao, Xue, Yap & Shen, 2018). The underlying adversarial strategy has proven powerful in making models trained on source data generalize to examples from a target domain, without having explicit target domain labels. In adversarial training, there are two networks, a generator and discriminator. A discriminator network classifies images produced by a generator network as real or fake, and the generator improves by learning to fool the discriminator. The application of this adversarial training can be seen for both natural (Chen, Li & Van Gool, 2018c; Hong, Wang, Yang & Yuan, 2018) images and medical images (Bateson, Kervadec, Dolz, Lombaert & Ayed, 2019; Kamnitsas *et al.*, 2017a). This technique is adapted to segmentation applications (Ghafoorian, Mehrtash, Kapur, Karssemeijer, Marchiori, Pesteie, Guttmann, de Leeuw, Tempany, van Ginneken et al., 2017; Javanmardi & Tasdizen, 2018; Vu, Jain, Bucher, Cord & Pérez, 2019; Zhang, David & Gong, 2017; Zhang, Miao, Mansi & Liao, 2018; Zou, Yu, Vijaya Kumar & Wang, 2018), where a segmentator network learns to produce accurate segmentation outputs for labeled source examples, and a discriminator which forces the segmentator to have a similar prediction for examples of both source and target domains. The segmentator and discriminator network are trained concurrently, while the segmentator aims to generalize in the feature space or in the output space. For classification tasks, Ganin & Lempitsky (2015); Long, Cao, Wang & Jordan (2015) proposed to reduce the distance between the source and target domain feature space. However for semantic segmentation, Tsai, Hung, Schulter, Sohn, Yang & Chandraker (2018) found robust performance when the output space is optimized. In addition to the above applications, adversarial training is shown to improve performance on source domain when there are few labels or no labels in the target domain (Bateson *et al.*, 2019; Dong, Cong, Sun, Yang, Xu & Ding, 2020; Dou, Ouyang, Chen, Chen & Heng, 2018; Kamnitsas *et al.*, 2017a).

The limited availability of manual annotated data is especially true for cortical surface analysis, where labelling thousands of nodes on a highly complex brain is very expensive. This also explains why the largest manually labelled brain surface dataset Mindboggle (Klein *et al.*, 2017), contains only 101 subjects. Moreover, another common problem of spectral graph convolution networks is that the models trained on the surfaces aligned to source domain usually fails to perform for samples aligned to other reference domain, i.e., the target domain. In this thesis, we address this limitation for brain surface analysis by adapting an adversarial domain adaptation method on surface graphs.

CHAPTER 2

GRAPH CONVOLUTIONS ON SPECTRAL EMBEDDINGS FOR CORTICAL SURFACE PARCELLATION

Karthik Gopinath[a], Christian Desrosiers[b], Herve Lombaert[c]

[a,b,c] Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

Paper published in *Journal of Medical Image Analysis* (MedIA), May 2019

Presentation

This chapter presents the article *"Graph Convolutions on Spectral Embeddings for Cortical Surface Parcellation"* Gopinath, Desrosiers & Lombaert (2018) submitted to the journal **MedIA** (Medical Image Analysis), sent on 31 July 2018, revised on 31 December 2018, and accepted for publication on 27 March 2019. The initial results were presented as a poster in **MISS** (Medical Imaging Summer School), 2018. Later, the work was published as an oral talk at **Med-NeurIPS** 2018 conference held in Montreal, QC. Additionally, the journal article was presented as a short paper poster at the MIDL conference (Medical Imaging with Deep Learning) held in London, UK. The objective of this article is to develop a learning and exploiting surface data directly across *multiple* surface domains with applications to cortical parcellation.

2.1 Introduction

Neuroimage analysis consists of studying functional and anatomical information over the brain geometry. Various aspects of the brain are investigated using different imaging modalities, such as magnetic resonance imaging (MRI) data. Structural MRI provides notably the geometry of the cortex. The thin outer layer of the brain cerebrum is of particular interest due to its vital role in cognition, vision, and perception. Statistical frameworks on surfaces are, therefore, highly sought for studying various aspects of the brain. For instance, variations in surface data can reveal new biomarkers as well as possible relations with disease processes (Arbabshirani *et al.*, 2017). The challenge consists of learning surface data over highly complex and convoluted surfaces and across different subjects.

The goal of separating the cerebral cortex into distinct regions based on structure or function is known as parcellation. Initially, automated parcellation techniques used clustering based on local regional statistics (Craddock, James, Holtzheimer III, Hu & Mayberg, 2012). For instance, a semi-supervised technique (Glasser, Coalson, Robinson, Hacker, Harwell, Yacoub, Ugurbil, Andersson, Beckmann, Jenkinson et al., 2016) delineated the cortical boundary from sharp changes in multimodal MRI data. Most research works use a cortical surface based feature to find surface correspondence. BrainVisa (Auzias, Lefèvre, Le Troter, Fischer, Perrot, Régis & Coulon, 2013; Cointepas, Geffroy, Souedet, Denghien & Rivière, 2010; Rivière, Régis, Cointepas, Papadopoulos-Orfanos, Cachia & Mangin, 2003) uses sulcal features defined by the cortical folding patterns to find correspondence between brain surfaces. Features like sulcal pits or sulcal lines (Auzias, Brun, Deruelle & Coulon, 2015; Lohmann, Von Cramon & Colchester, 2007) are other existing features used for estimating surface correspondences. Conventional approaches typically rely on geometrical simplifications, such as spherical inflation and slow mesh deformations (Styner, Oguz, Xu, Brechbühler, Pantazis, Levitt, Shenton & Gerig, 2006; Tustison, Cook, Klein, Song, Das, Duda, Kandel, van Strien, Stone, Gee & Avants, 2014; Yeo, Sabuncu, Vercauteren, Ayache, Fischl & Golland, 2010), a popular but costly process. For instance, the widely used FreeSurfer (Fischl *et al.*, 2004a) takes around 3 hours to parcellate brain surfaces by slowly deforming brain models towards labeled atlases.

Convolutional Neural Networks (CNNs) (Lecun, Bottou, Bengio & Haffner, 1998) have the potential to offer a drastic speed advantage over traditional surface-based methods. CNNs are mostly used in neuroimage analysis for segmentation (Wachinger *et al.*, 2017) or finding structural abnormalities (Valverde *et al.*, 2017). The network architecture is either fixed

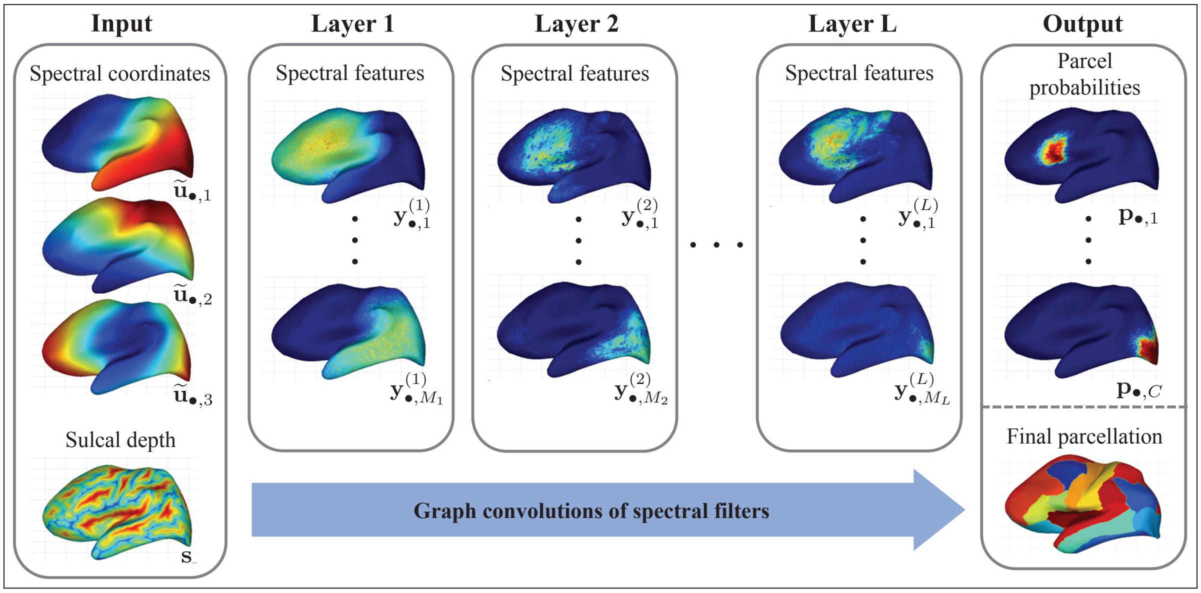


Figure 2.1 Overview of our spectral graph convolution algorithm – Graph convolutions of spectral filters are applied sequentially to process cortical surface data. On the left are inputs: Surface data, such as sulcal depth, s, and aligned spectral coordinates, $\widetilde{\mathbf{u}}$. In the middle are the learned spectral features, $\mathbf{y}$, found in each layer. On the right are: Predicted parcel probabilities, $\mathbf{p}$, given by the softmax and the final surface parcellation. Coloring represents the pointwise value of respective maps from low (blue) to high (red) values.

for various segmentation applications (Ronneberger *et al.*, 2015) or tailored to particular problems (Kamnitsas *et al.*, 2017b). Fundamentally, current statistical frameworks exploit spatial information mostly derived from the Euclidean domain, for instance, based on vector fields, image or volumetric coordinates (Dolz, Desrosiers & Ben Ayed, 2017; Hua, Hibar, Ching, Boyle, Rajagopalan, Gutman, Leow, Toga, Jack, Harvey, Weiner, Thompson & Alzheimer's Disease Neuroimaging Initiative, 2013; Kamnitsas *et al.*, 2017b; Zhang & Davatzikos, 2011). Such information is highly variable across brain geometries and severally hinders the training of modern machine learning algorithms.

Geometric deep learning (Bronstein *et al.*, 2017) recently proposed to use convolutional filters on irregular graphs. To handle the neural network on a graph, Scarselli, Gori, Tsoi, Hagenbuchner & Monfardini (2009) proposes to map and learn graph data in a high-dimensional Euclidean space. Lately, (Bruna *et al.*, 2014b) formulates the convolution theorem from Fourier space to spectral domains over graphs. Chebyshev polynomials are also used to avoid the explicit computation of graph Laplacian eigenvectors (Defferrard, Bresson & Vandergheynst, 2016). The main concern of these methods is their inability to compare surface data across *different* surface domains (Bronstein, Glashoff & Loring, 2013; Eynard, Kovnatsky, Bronstein, Glashoff & Bronstein, 2015; Kovnatsky, Bronstein, Bronstein, Glashoff & Kimmel, 2013; Ovsjanikov, Ben-Chen, Solomon, Butscher & Guibas, 2012). Laplacian eigenbases are indeed incompatible across multiple geometries. Alternatively, Boscaini, Masci, Rodolà & Bronstein (2016); Masci, Boscaini, Bronstein & Vandergheynst (2015) proposed a graph convolution approach in the spatial domain. These approaches map local graph information onto geodesic patches and use conventional spatial convolution as template matching. For instance, Monti *et al.* (2017) obtains geodesic patches with local parametric constructions of tangent planes to the surface. Another prominent spatial approach Veličković *et al.* (2018) proposes to include self-attentional layers in which neighborhoods are used to avoid an explicit computation of a graph Laplacian. This attentional approach reduces to a particular formulation of Monti *et al.* (2017). A related work (Simonovsky & Komodakis, 2017) also conditions convolutional filter weights on specific edge labels over neighborhoods rather than on graph nodes. Applications of graph convolution networks in neuroimaging remain yet limited. Existing work includes the use of graph convolutions over population graphs for predicting brain disorders and learning distance metrics in functional brain networks (Ktena, Parisot, Ferrante, Rajchl, Lee, Glocker & Rueckert, 2017; Parisot, Ktena, Ferrante, Lee, Guerrero, Glocker & Rueckert, 2018). A recent work (Cucurull *et al.*, 2018) proposes to parcellate the cerebral cortex into three parcels using an attention-based method (Veličković *et al.*, 2018). Brain meshes are, however, constrained within a unique graph structure, limiting all meshes to use the same mesh geometry. Fundamentally, these methods either lack the capability to process multiple surface domains (Bronstein *et al.*, 2013; Eynard *et al.*, 2015; Kovnatsky *et al.*, 2013; Ovsjanikov *et al.*, 2012) or have spatial representations of surface data defined in a Euclidean space, which ignore the complex geometry of the surface. They rely, for instance, on polar representations of mesh vertices (Boscaini *et al.*, 2016; Masci *et al.*, 2015; Monti *et al.*, 2017; Veličković *et al.*, 2018).

This paper leverages recent advances in spectral graph matching to transfer surface data across aligned spectral domains (Lombaert, Arcaro & Ayache, 2015a). The transfer of spectral coordinates across domains provides a robust formulation for spectral methods that naturally handles differences across Laplacian eigenvectors, including sign flips, ordering, and mixing of eigenvectors in higher frequencies. This spectral alignment strategy was exploited to learn surface data (Lombaert *et al.*, 2015b) within the random forest framework. Spectral Forests are operating in a spectral domain and use the first spectral coordinates as well as sulcal depth of each cortical point. This approach is, however, limited to only pointwise information, ignoring local patterns within surface neighborhoods. Our approach consists of leveraging spectral coordinates within graph convolutional networks to bridge a gap between learning algorithms and geometrical representations. To the best of our knowledge, this is the first attempt at intrinsically learning surface data via spectral graph convolutions in neuroimaging. This novel approach enables a *direct* learning of surface data across compatible surface bases by exploiting spectral filters over intrinsic representations of surface neighborhoods.

The main contributions of our work are:

- A novel *spectral* graph convolutional approach for cortical parcellation;
- A *direct* learning of surface data using trainable spectral filters over surface embeddings;
- The training of spectral filters across *multiple* mesh geometries of various graph structures;
- The leverage of spectral filters to exploit *local* patterns of data within surface neighborhoods;
- An evaluation on the *largest* publicly available dataset of manually labeled brain surfaces (Klein *et al.*, 2017);

- An improved state-of-the-art performance for cortical surface parcellation with graph convolutions;

In this work, we propose a surface learning algorithm. We illustrate the learning capabilities of this approach with an application to brain parcellation. We choose cortical parcellation since it provides established benchmarks with publicly available datasets of manual labelings. The validation over the largest publicly available dataset of manually labeled brain surfaces (Klein *et al.*, 2017), with 101 subjects, demonstrates a significant improvement in using spectral graph convolutions over Euclidean approaches. This change of paradigm indeed improves the parcellation accuracy when using graph convolutions, from a Dice score of 53% to 85%. Our approach is at least at par with the well established FreeSurfer algorithm (Fischl *et al.*, 2004a) when benchmarking over a large dataset (Klein *et al.*, 2017), while gaining a drastic speed improvement in the order of seconds. The next section details the fundamentals of our spectral approach, followed by experiments evaluating the impact of our spectral strategy over standard Euclidean approaches for graph convolutions.

2.2 Method

An overview of the proposed method is shown in Fig. 2.1. In a first step, each cortical surface is modeled as a graph. Spectral decomposition is then applied on these graphs to capture the intrinsic geometry of brain surfaces and embed this information in a low-dimensional feature space (Lefèvre, Pepe, Muscato, De Guio, Girard, Auzias & Germanaud, 2018; Lombaert *et al.*, 2015a). Subsequently, the transfer of surface data between spectral embeddings enables graph convolution networks to process cortical data across multiple mesh domains. This is implemented with a realignment of spectral embeddings. Finally, cortical parcellation is performed by learning spectral filters over realigned spectral coordinates and cortical features like the sulcal depth. Dense connections (Huang, Liu, van der Maaten & Weinberger, 2017) improve convergence by propagating information from the initial layers to output layers. We

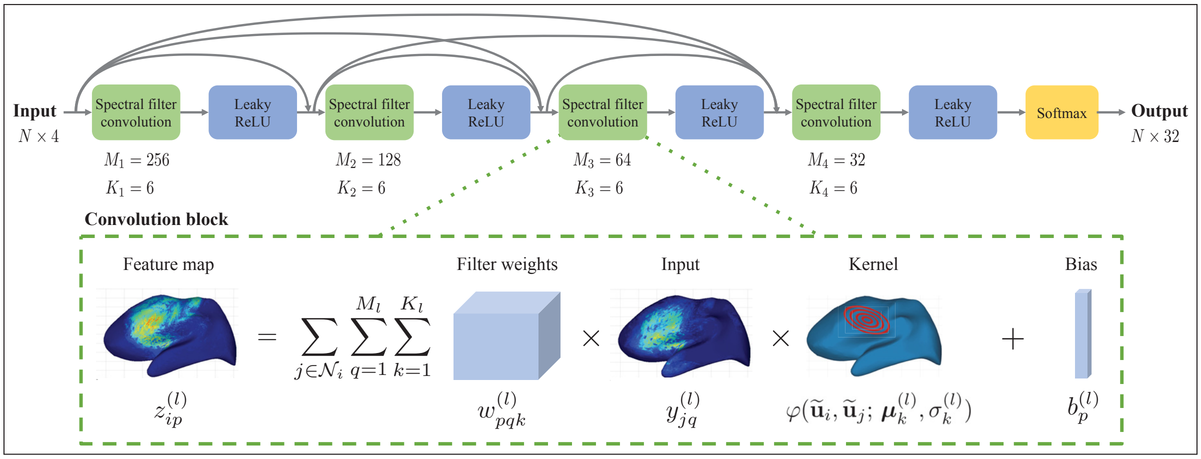


Figure 2.2 Overview of the network architecture – Dense connections are used among successive layers constituted with graph convolutions of learned spectral filters and leaky ReLU activations. Weights (w), biases (b), and parameters of our spectral filters (μ, σ) are learned via back-propagation. A final softmax function produces parcel probabilities (p) on the brain surface.

therefore use dense connections among successive graph convolution layers. The overview of the network architecture is shown in Fig. 2.2.

2.2.1 Spectral embedding of brain graphs

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a brain graph defined with node set $\mathcal{V}$, such that $N = |\mathcal{V}|$, and edge set $\mathcal{E}$. Each node i has a feature vector $\mathbf{v}_i = (\mathbf{x}_i, \mathbf{s}_i)$ composed of 3D spatial coordinates $\mathbf{x}_i$ and surface data features $\mathbf{s}_i$. Various features could be considered to model the local geometry of the cortical surface, including mean curvature, average convexity, and cortical thickness (Fischl *et al.*, 2004a; Li, Wang, Shi, Gilmore, Lin & Shen, 2015). In this work, we use sulcal depth since the boundaries of several regions in anatomical parcellation protocols typically follow such sulcal features (Destrieux, Fischl, Dale & Halgren, 2009).

We map $\mathcal{G}$ to a low-dimensional subspace using the eigencomponents of the normalized graph Laplacian operator $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{A}$ is the weighted adjacency matrix and $\mathbf{D}$ is the diagonal degree matrix (Chung, 1997). Here, we define the weight between two adjacent nodes in terms of node affinity (Grady & Polimeni, 2010), such as the inverse of their Euclidean distance: $a_{ij} = \left(\|\mathbf{x}_i - \mathbf{x}_j\|_2 + \epsilon\right)^{-1}$ where ϵ is a small constant to avoid a zero-division. Let $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ be the eigendecomposition of the normalized Laplacian matrix $\mathbf{L}$. Since the most relevant characteristics of the embedded surface are captured by the principal spectral components of $\mathbf{L}$, following Lombaert *et al.* (2015b), we limit the decomposition to the $d = 3$ first smallest non-zero eigenvalues of $\mathbf{L}$. We then obtain the normalized spectral coordinates of nodes as the rows of matrix $\widehat{\mathbf{U}} = \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}$.

Because the spectral embedding of $\mathbf{L}$ is only defined up to an orthogonal transformation, we need to align spectral representations of different brain graphs to an arbitrary reference. Denote as $\widehat{\mathbf{U}}^{(0)}$ the normalized spectral embedding of this reference, we align an embedding $\widehat{\mathbf{U}}$ to $\widehat{\mathbf{U}}^{(0)}$ with an iterative method based on the ICP algorithm (Lombaert *et al.*, 2015a). In this method, each node $i \in \mathcal{V}$ is mapped to its nearest reference node $\pi(i) \in \mathcal{V}^{(0)}$ in the embedding space via a nearest neighbor search. The optimal transformation $\mathbf{R}$ between matched nodes is then obtained by solving a Procrustes analysis problem. Let $\widehat{\mathbf{u}}_i$ be the normalized spectral coordinates of node i, the overall alignment process can thus be formulated as:

$$\underset{\pi,\mathbf{R}}{\arg\min} \sum_{i=1}^{N} \left\|\mathbf{R}\,\widehat{\mathbf{u}}_i - \widehat{\mathbf{u}}^{(0)}_{\pi(i)}\right\|_2^2. \tag{2.1}$$

We solve this problem by updating the node correspondence mapping π and the transformation $\mathbf{R}$ as described above, until convergence.

2.2.2 Graph convolution on surfaces

We start by presenting the standard CNN model for rigid grids and then explain how this model can be extended to an arbitrary geometry. Let $\mathbf{Y}^{(l)} \in \mathbb{R}^{N \times M_l}$ be the input feature map at convolution layer l of the network, such that $y^{(l)}_{iq}$ is the q-th feature of the i-th input node. The network input thus corresponds to $\mathbf{Y}^{(1)}$. Assuming a 1D grid, the output feature map of layer l is given by $y_{ip}^{(l+1)} = f(z_{ip}^{(l)})$ with:

$$z_{ip}^{(l)} = \sum_{q=1}^{M_l} \sum_{k=-K_l}^{K_l} w_{pqk}^{(l)} \cdot y_{i+k,q}^{(l)} + b_p^{(l)}. \tag{2.2}$$

Here, $w_{pqk}^{(l)}$ are the convolution kernel weights, $b_p^{(l)}$ the bias weights of the layer, and f is a non-linear activation function, for instance, the sigmoid or rectified linear unit (ReLU) functions.

To extend this fixed-grid formulation to a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, we denote as $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$ the neighbors of node $i \in \mathcal{V}$. A generalized convolution operation can then be defined as:

$$z_{ip}^{(l)} = \sum_{j \in \mathcal{N}_i} \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)} \cdot y_{jq}^{(l)} \cdot \varphi_{ij}(\boldsymbol{\Theta}_k^{(l)}) + b_p^{(l)}, \tag{2.3}$$

where $\varphi_{ij}(\boldsymbol{\Theta}_k)$ is a symmetric kernel with parameters $\boldsymbol{\Theta}_k$. In Monti *et al.* (2017), this kernel is defined on a tangent plane of a mesh at node i and is parameterized using polar coordinates. Learning is however constrained to a *single* graph structure, which hinders the application of convolutions across multiple graphs.

2.2.3 Learning across multiple mesh geometries

To learn surface data across multiple graphs, we leverage the spectral transfer of information across spectral embeddings. The transformation $\mathbf{R}$ of Eq. (2.1) is first used to obtain the aligned spectral coordinates with $\widetilde{\mathbf{U}} = \mathbf{R}\,\widehat{\mathbf{U}}$. Convolution kernels φ is then defined in the common spectral domain:

$$z_{ip}^{(l)} = \sum_{j \in \mathcal{N}_i} \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)} \cdot y_{jq}^{(l)} \cdot \varphi(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j;\, \boldsymbol{\Theta}_k^{(l)}) + b_p^{(l)}. \tag{2.4}$$

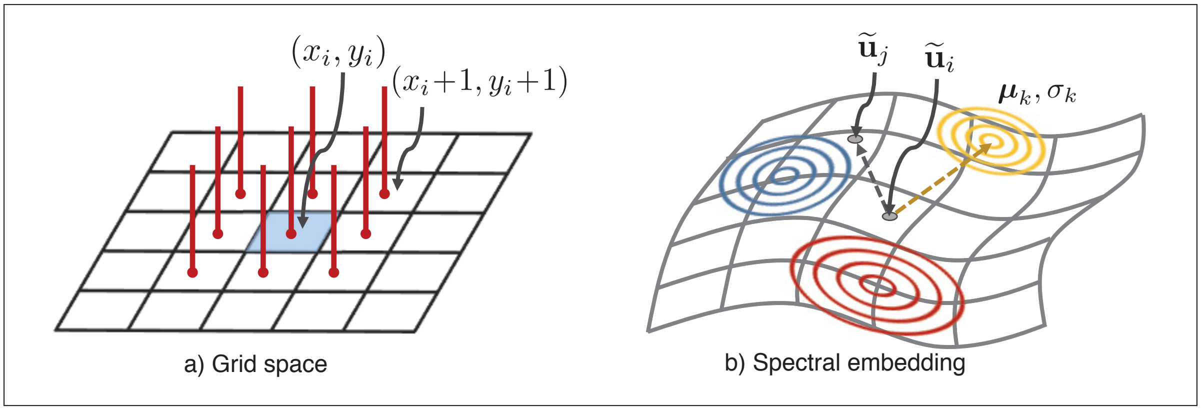


Figure 2.3 Illustration of the convolution operation – (a) Standard convolution on grids. (b) Geometric convolutions using three Gaussian kernels on a spectral embedding. Operations on grids can be shown to be a special case of operations on surfaces.

While any symmetric kernel can be used, in this work, we set φ as the Gaussian kernel with mean (or *offset*) $\boldsymbol{\mu}_k$ and variance (or *bandwidth*) σ_k:

$$\varphi(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j; \boldsymbol{\mu}_k, \sigma_k) \;=\; \exp\big(- \|(\widetilde{\mathbf{u}}_j - \widetilde{\mathbf{u}}_i) - \boldsymbol{\mu}_k\|_2^2 \,/ 2\sigma_k^2 \big). \tag{2.5}$$

In an image domain, the neighborhood structure is regular, often organized in a lattice with fixed edge lengths between neighboring pixels. However, in a graph embedding, neighborhoods can have arbitrary structures with different edge lengths across the embedding. The continuous spectral domain embeds the geometric information of the graph. We define learnable kernels in a spectral domain relative to node i in order to capture the neighborhood information. This is thus the reason for subtracting $\widetilde{\mathbf{u}}_i$ in Eq. (2.5). Keeping kernels relative to node i allows the application of convolutions over a continuous space and reusing the same kernel parameters across different nodes. This fundamental change from Euclidean space to the spectral domain enforces the learning process to be geometry-aware.

The difference between the proposed graph convolution and standard convolutions over a fixed grid is illustrated in Fig. 2.3. Grid-based convolutions can be seen as a special case of our graph convolutions, for which the kernel offsets $\boldsymbol{\mu}_k$ are positioned regularly on the grid and the bandwidth $\sigma_k \to 0$. In addition to extending standard convolutions to irregular grids in continuous space, the proposed formulation can also model filter responses at different scales by varying σ_k. Moreover, since kernel parameters $\boldsymbol{\mu}_k$ and σ_k are learned directly from training data, instead of being defined during architecture design, our formulation can better adapt to the task at hand. For instance, kernels with small bandwidth can be learned to recognize thinner cortical structures, while large bandwidth kernels can be learned to model broader regions. Finally, the proposed strategy avoids the use of tangent planes, polar pseudo-coordinates (Monti *et al.*, 2017) or the costly computation of geodesic distances (Boscaini *et al.*, 2016; Masci *et al.*, 2015).

Using the formulation of Eq. (2.4), we define a fully-convolutional network whose input at node i is given by $\mathbf{y}_i^{(0)} = (\widetilde{\mathbf{u}}_i, s_i)$, where $\widetilde{\mathbf{u}}_i$ is the aligned spectral coordinates of i and s_i is the sulcal depth at this node. The output layer of the network has a size corresponding to the number of parcels to be segmented, 32 in our case. Leaky ReLU is applied after each layer to obtain filter responses: $y_{ip}^{(l)} = \max(0.01 z_{ip}^{(l)}, z_{ip}^{(l)})$. Since the parcels to segment are mutually exclusive, we use a softmax operation after the last graph convolution layer to obtain the parcel probabilities of each node. The softmax function is given by $\frac{exp(y_{ip}^{(l)})}{\sum_q exp(y_{iq}^{(l)})}$. Finally, the weighted cross-entropy is employed as output loss function:

$$E(\boldsymbol{\Theta}) \;=\; -\sum_{i=1}^{N}\sum_{c=1}^{C} \omega_c \cdot s_{ic} \cdot \log p_{ic}(\boldsymbol{\Theta}), \tag{2.6}$$

where $\boldsymbol{\Theta} = \{w_{pqk}^{(l)}, b_p^{(l)}, \boldsymbol{\Theta}_k^{(l)}\}$ are the trainable network parameters, $p_{ic}(\boldsymbol{\Theta})$ is the output probability for node i and parcel label c, and s_{ic} is a one-hot encoding of the reference segmentation. The weights ω_c compensate for the size difference of parcels, and are set inversely proportional to their surface areas such that the larger and smaller parcels have similar importance. This loss is minimized by back-propagating the error using standard gradient descent optimization.

2.3 Results

We now validate our spectral learning approach. To do so, we benchmark our performance using the largest publicly available dataset of manually labeled brain surfaces, Mindboggle (Klein *et al.*, 2017). It contains 101 subjects collected from different sites, with cortical meshes varying from 102K to 185K vertices. Each brain surface includes 32 manually labeled parcels. The experiments are carried out on an i7 desktop computer with 16GB of RAM and a Nvidia Titan X GPU. First, we evaluate the influence of different parameters in our learning framework. Second, we highlight the effect and advantages of spectral alignment in this framework. Finally, we assess the improvement in accuracy of learning frameworks when directly operating in a spectral domain rather than a conventional Euclidean domain.

2.3.1 Model selection

The hyper-parameters in our formulation are the number of graph convolution layers, L, in the fully-convolutional network and the number of Gaussian kernels, K, in each layer. To evaluate the effect of these parameters, we first set the size of the output layer to be equal to the number of parcels, 32 in our case. We measure the change in performance when increasing the number of layers from 1 to 4. Dense connections are removed in order to benchmark performance in a same controlled experimental setting. Along the layers of our architecture, the size of feature maps is reduced by two between two consecutive layers. This is to focus information towards the final parcellation and to limit memory usage. We also evaluate the performance when changing the number of kernels from 1 to 7. Our goal is to study how performance improves with an increasing number of layers and kernels. The total number of trainable parameters in a layer l that has a feature map of size M_l and K kernels is given by $K\left(M_{l-1} + M_{l-1} \times M_l\right) + M_l$. This indicates that the number of trainable parameters grows with the size of our network architecture. The model becomes computationally expensive in terms of memory usage for architecture beyond 4 layers and 7 kernels. Using relatively large sized brain meshes of the Mindboggle

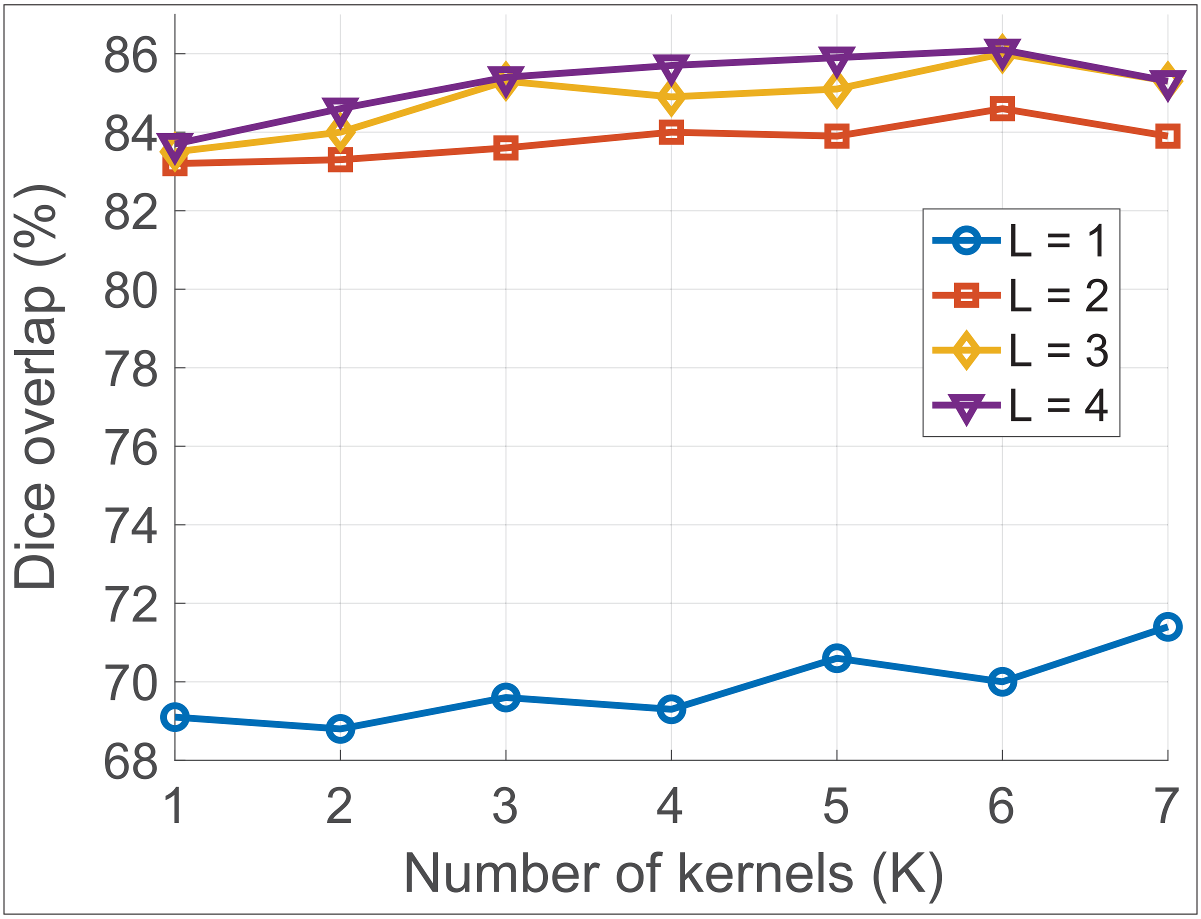


Figure 2.4 Experiments for model selection – Each line indicates the segmentation accuracy in terms of Dice scores on the test split for different architectural models. It is observed that performance improves when the number of layers, L, increases, but quickly reaches a plateau. Performance also increases with the number of kernels, K. A peak is observed with 6 kernels and 4 layers. This configuration requires about 10GB of RAM. Increasing the model complexity would unnecessarily burden memory usage.

dataset, we use random splits for training, validation, and testing in a 70-10-20% ratio. Each evaluated architecture is trained for 250 epochs on the train split. We observe an increase in segmentation accuracy in terms of Dice score when the number of kernels, K, increases. However, for the same number of kernels, adding more layers burdens the model complexity, while accuracy increases but stagnates from 2 to 4 layers. This is shown in Fig. 2.4.

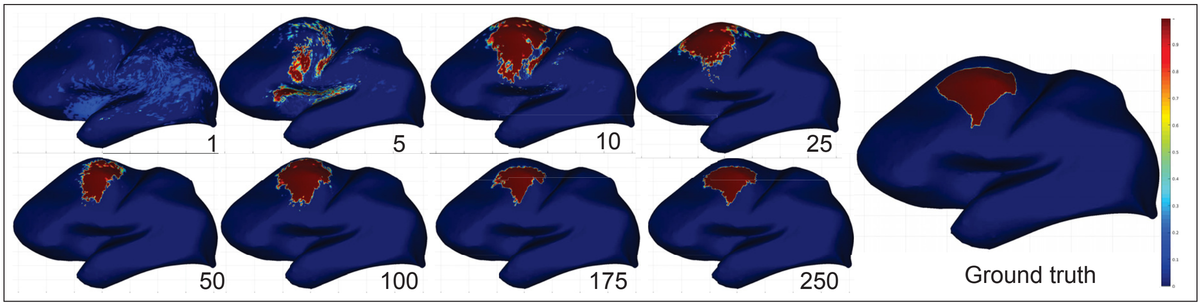


Figure 2.5 Evolution of learning algorithm – The prediction of a particular parcel over multiple epochs is shown. A coarse to fine refinement of the parcel region is observed. After training, the predicted parcel probability corresponds to the ground truth parcel, shown on the right.

Table 2.1 Robustness to reference across all parcels – The average dice percentage obtained after separate training and testing with 5 references. The last column provides the mean and standard deviation of the results across all 35 parcels tested with all 5 references.

Ref_1	Ref_2	Ref_3	Ref_4	Ref_5	**Mean**
86.62 ± 1.72	86.20 ± 1.56	86.42 ± 1.73	86.26 ± 1.67	86.52 ± 1.75	$\mathbf{86.40 \pm 0.17}$

This experiment indicates that $L = 4$ with $K = 6$ would be the optimal hyper-parameter values. The model has a total number of 264, 768 trainable parameters and takes up to 10GB of GPU memory. The trainable parameters for our dense version doubles and takes up to 11GB of shared GPU memory. We similarly train our dense model for 250 epochs. The best performing model on the validation set is used for testing. Fig. 2.5 illustrates the evolution of the learning algorithm over iterations when classifying one parcel. We use this dense model for the rest of our experiments.

2.3.2 Spectral alignment

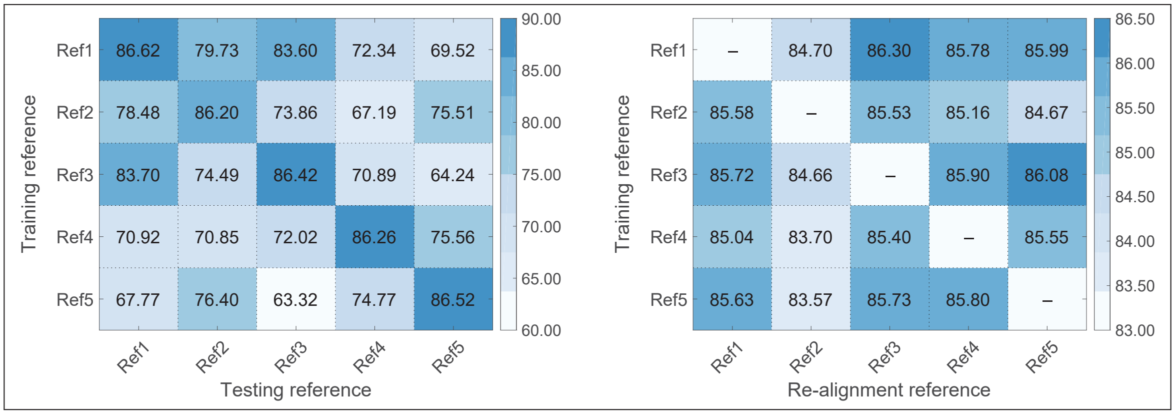
Our contribution is to operate in a geometry-aware spectral domain. This is enabled by aligning spectral embeddings across various mesh domains. We now evaluate the effect of a spectral alignment when learning graph convolution kernels. We align the spectral representations of different brain graphs to an arbitrary reference from the dataset.

First, we verify independence of our method with respect to the choice of a reference for alignment. To do so, we train and test our algorithm with 5 different reference brains, where the whole dataset is aligned to a reference. The evaluation of our algorithm on these 5 different references shows a performance having an average dice score of 86.4% and a standard deviation of 0.17% (Table 2.1). This indicates that our learning algorithm is robust to the choice of reference.

Second, we evaluate the impact of aligning spectral embeddings in learning graph convolution kernels. When both training and testing sets are aligned towards one same reference, the trained model yields an accuracy of 86.62% in terms of Dice overlap. However, when training and testing sets are both aligned towards differing references, $\text{Ref}_{\text{training}}$ and $\text{Ref}_{\text{testing}}$, the performance drops to 79.73%. This may be expected since both training and testing sets are expressed using differing spectral domains. To solve this discrepancy, our methodology consists of realigning the testing set towards one unique spectral domain, for instance, using $\text{Ref}_{\text{training}}$. In such case, the performance on realigned embeddings is improved to 84.7%. To evaluate the effect of varying references between training and testing sets, we iterate over all possible combinations, summarized in Fig. 2.6. It is observed that if spectral alignment is not present between the training and testing set, the classification accuracy degrades, while a spectral realignment of the testing set brings back the accuracy to initial scores.

2.3.3 Comparison with the state-of-the-art

We now compare our method with state-of-the-art approaches in learning graph-structured methods. First, we show the limitations of working in the Euclidean domain with the Random Forest method (Lombaert *et al.*, 2015b) as well as the latest approaches of graph convolution networks (Monti *et al.*, 2017). Second, we show the advantage of changing the paradigm in graph learning frameworks from operating in a conventional Euclidean domain towards a spectral domain. This is enabled by our transfer of spectral embeddings across brain surface



	Ref1	Ref2	Ref3	Ref4	Ref5
Ref1	86.62	79.73	83.60	72.34	69.52
Ref2	78.48	86.20	73.86	67.19	75.51
Ref3	83.70	74.49	86.42	70.89	64.24
Ref4	70.92	70.85	72.02	86.26	75.56
Ref5	67.77	76.40	63.32	74.77	86.52

	Ref1	Ref2	Ref3	Ref4	Ref5
Ref1	–	84.70	86.30	85.78	85.99
Ref2	85.58	–	85.53	85.16	84.67
Ref3	85.72	84.66	–	85.90	86.08
Ref4	85.04	83.70	85.40	–	85.55
Ref5	85.63	83.57	85.73	85.80	–

Figure 2.6 Robustness to the choice of reference – (Left) Dice score performance of two spectral models trained and tested using different aligned references. Row-column (i, j) provides the score of a using model trained via reference i, and tested via reference j. If references differ, scores degrade. This illustrates the current limitation of current graph convolutional approaches. (Right) Dice score performance of models when references are aligned. Row-column (i, j) indicates that difference references are used during trained and testing. The higher scores in off-diagonal experiments indicates that a realignment of spectral embeddings is essential to exploit multiple mesh domains.

Table 2.2 Comparisons with graph learning approaches – Average dice overlaps (in %) over 32 parcels of 101 subjects are shown along classification accuracy (in %), and average Hausdorff distances (in millimeters).

Method	**Dice overlap (%)**	**Accuracy (%)**	**Avg. Hausdorff (mm)**
Euclidean Forest	45.87 ± 8.74	49.26 ± 8.32	4.97 ± 1.11
GC on Euclidean	50.78 ± 10.78	54.24 ± 10.33	5.82 ± 1.66
Spectral Alignment	77.67 ± 3.65	81.87 ± 3.39	2.87 ± 0.47
Spectral Forest	79.89 ± 2.62	81.94 ± 2.54	1.97 ± 0.40
FreeSurfer	84.39 ± 1.91	85.19 ± 1.98	2.11 ± 0.29
Ours	85.37 ± 2.36	86.97 ± 2.43	1.75 ± 0.35
Ours + MRF	**86.61 ± 2.45**	**88.08 ± 2.47**	**1.66 ± 0.44**

domains. Finally, we assess the improvement of exploiting neighborhoods of surface data versus the pointwise Spectral Forest method (Lombaert *et al.*, 2015b), as well as a comparison with the established FreeSurfer algorithm (Fischl *et al.*, 2004a).

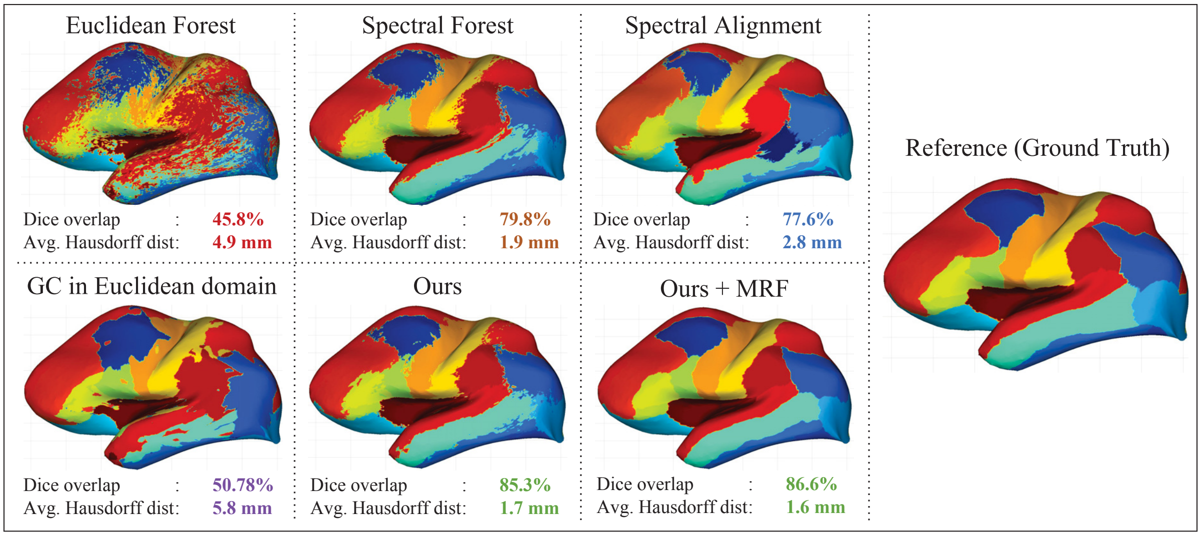


Figure 2.7 Cortical parcellation comparison – (*First column, Left*) Learning with Euclidean coordinates: yields low Dice score (45.8% with Random Forests, 50.8% with graph convolutions) and inconsistent boundaries (Hausdorff distance of 4.9-5.8mm). (*Second column, Middle*) Learning with Spectral coordinates: improves Dice score (79.8% with Spectral Forests, 85.3% with our Spectral convolutions) and boundary regularity (1.9-1.7mm). (*Third column, Top*) A pure spectral alignment without learning yields a Dice score of 77.6%. This is used as a benchmark to assess improvement in learning strategies. (*Third column, Bottom*) The parcel probability maps generated with our spectral filters could be further refined with an MRF regularization, leading to an improvement in boundary regularity (1.6mm) and Dice score (86.6%). (*Right*) Reference ground truth for comparison purposes. Brain surfaces are inflated for visualization.

We train and test all methods on the entire dataset with a 5-fold cross-validation. We train a random forest with 50 trees on 3D spatial location and sulcal depth, which we name Euclidean Forest, as in Lombaert *et al.* (2015b). For comparison, we also train a graph convolution network similarly to Monti *et al.* (2017) in the Euclidean domain, with 3D spatial location and sulcal depth. The architecture remains the same as described earlier, as in Fig. 2.2. The average Dice overlap across all parcels in our dataset is 45.87% with Euclidean Forests. Graph convolution networks in the Euclidean space has an average Dice overlap of 50.78%, which is an increase of 4.9%.

To put in perspective, a pure spectral alignment of brain surfaces yields aligned parcellations that have an average Dice overlap of 77.67% over all pairs of possible brains. This pure alignment

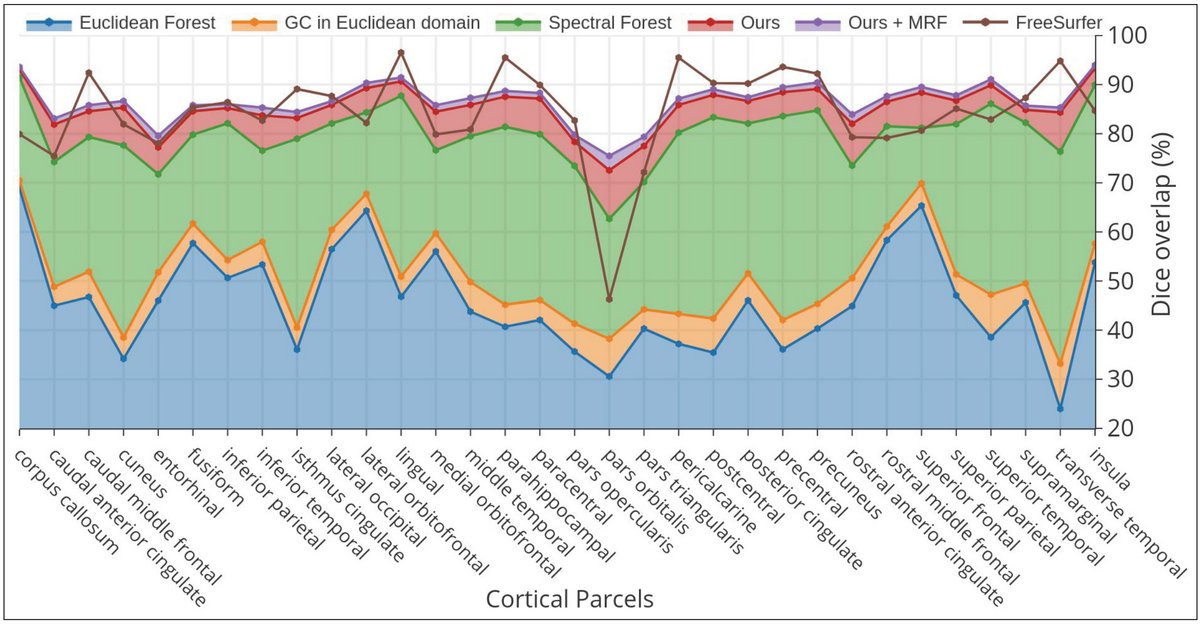


Figure 2.8 Performance evolution of different methods for each individual parcels – Dice scores for all 32 cortical parcels across the dataset when learning with: (*Blue*) Euclidean Forest, (*Orange*) Graph convolutions in the Euclidean domain, (*Green*) Spectral Forest with pointwise information, (*Red/Ours*) Graph convolutions in a spectral domain, exploiting neighborhood information, (*Purple/Ours+MRF*) Final MRF refinement of our spectral maps, and (*Brown*) FreeSurfer provided for comparison. Improvements are consistent across all 32 parcels. The first leap in accuracy (*Orange area*, +11%) corresponds to an improvement from using convolutional networks over random forests. The second leap (*Green area*, +58%) corresponds to an improvement from learning in a spectral domain rather than Euclidean. The third leap (*Red area*, +7%) corresponds to the extra improvement of exploiting spectral neighborhoods when learning spectral convolutional filters. The fourth leap (*Purple area*, +1%) indicates the effect of regularizing final parcel probability maps with MRFs.

process does not involve learning of surface data while producing scores 16.9% superior to the previous Euclidean learning approaches. This indicates the benefits of operating in a spectral domain instead of the conventional Euclidean domain.

The effect of learning over spectral domains is assessed using, first, pointwise information in the Random Forest framework and in graph convolutional networks. Input for all random forests consists of sulcal depth and the first three spectral coordinates. This is referred to the Spectral Forest, similarly to Lombaert *et al.* (2015b), and yields an average Dice overlap of 79.89%. It is important to note that Spectral Forests learn over pointwise information only. Our graph convolutional network benefits from exploiting neighborhood information of surface data. The trained kernels on spectral embeddings yield an average Dice overlap of 85.37%. These results are summarized in Table 2.2. Fig. 2.7 shows that indeed learning using spectral method produces an improved parcellation quantitatively. The qualitative results from our algorithm look similar to the manual parcellation.

As an illustration of further refinement, we use a Markov Random Field (MRF) regularization for our method. We apply a standard graph cut algorithm (Boykov & Kolmogorov, 2004) with minus-log parcel probabilities as unary potentials and the Potts model for defining binary potentials. MRF regularization further improves the overall classification accuracy from 85.37% to 86.62.1%. The Spectral Forest parcellates the brain with an average Dice overlap of 79.8%. With an MRF as post-processing over the prediction of Spectral Forest, the Dice overlap improves to 83.7%, still lower than our approach without MRFs. The improvement of 3.9% in Dice overlap is seen with the use of neighborhood information from MRF. We see an increase of 1.3% in terms of Dice overlap when we use MRF with our approach. We can observe a similar improvement regarding average Hausdorff distances, with a reduction of distance error from 2.1 mm to 1.75 mm (Table 2.2). However, our graph convolution based approach has higher performance of 2.7% average Dice overlap over the Spectral Forest + MRF. A closer look at the performance scores for each parcel (Fig. 2.8) also reveals a general improvement when exploiting neighborhoods (Our method) over pointwise surface data (Spectral Forest). This is a 34.59% improvement (Table 2.2) over learning in the standard Euclidean domain.

2.4 Discussion and Conclusion

This paper presented a novel framework for learning surface data via graph convolutions of spectral filters. The algorithm leverages recent advances in spectral matching to enable the comparison of surface data across multiple surface domains. Our experiments illustrated the benefits of our approach with an application to cortical surface parcellation. This is a particularly challenging problem where current graph convolutional approaches remain limited by the inability to compare surface data across brain geometries. This typically results in spatial irregularities of parcel boundaries as illustrated in Fig. 2.7.

Shifting graph convolutions into a spectral domain endows the learning process with a geometry-aware representation of surface data. This strategy reveals that the use of spectral features improves a classification from a 50% Dice score in a conventional Euclidean domain to an 85% Dice score in a spectral domain. A performance gain is also noted when using a graph convolutional network instead of a standard random forest, from 45% to 50% when learning from a conventional representation of spatial information. Our experiments further indicate that an extra improvement is also gained by exploiting spectral neighborhoods. Fig. 2.8 indeed exhibits a major performance leap when leveraging spectral features over Euclidean features. This corresponds to the green area in Fig. 2.8, from 50% to 79% – a 29% improvement. The next leap in the graphic indicates an improvement due to exploiting spectral neighborhoods, from pointwise Spectral Forests (Lombaert *et al.*, 2015b), where no neighborhood can be exploited with random forests, to our graph convolutional approach exploiting spectral neighborhoods. It is also worth noting that our graph convolutional approach uses only the first three principal components whereas the Spectral Forests (Lombaert *et al.*, 2015b) considers five principal components to perform the same task. This is the red area in Fig. 2.8, from 79% to 85%, across all parcels in our dataset – an extra 7% improvement. These results confirm that exploiting a spectral domain provides a significant gain in performance, 29%, and exploiting convolutions over spectral neighborhoods provide an additional 7% improvement.

The experiments used the largest publicly available dataset of manually labeled brain surfaces (Klein *et al.*, 2017). The performance of our spectral strategy is comparable to the state-of-the-art approaches for cortical parcellation. It reduces, however, the computation time by an order of magnitude. The Spectral Forest approach requires 23 seconds to run one pass over all 50 trees, yielding an accuracy of 79.89% ± 2.62%. Our method requires 3 seconds to run one forward-pass on a trained network, yielding an accuracy of 85.37% ± 2.36%. Both models roughly take 15 seconds to obtain the spectral coordinates and align to a reference brain. This is an 83% improvement in computation time. Using an additional MRF regularizer, as is often used in network approaches, brings up accuracy to 86.61% ± 2.45%. The parcellation process of FreeSurfer, starting from a brain surface, requires 3 hours of computation and yields an accuracy of 84.39% ± 1.91%. It is to be also noted that in the protocol of the Mindboggle dataset, annotations by experts are, in effect, manual corrections from FreeSurfer parcellations (Klein *et al.*, 2017). This creates a positive bias for FreeSurfer results. Our claim in our experiments is not necessarily a superiority of our approach, but to rather provide a parcellation accuracy that is at least equivalent to FreeSurfer.

The advantage of using a spectral method is, on one hand, computational, by providing parcellation in seconds rather than hours, and on the other, methodological, by opening up a new learning strategy for processing cortical surface data. The technical contribution leveraged recent work on transfer of spectral bases across brain surface domains (Lombaert *et al.*, 2015a,1). This enables the learning of spectral convolution filters across *multiple* brain geometries. This overcomes a major limitation in current graph convolutional approaches (Boscaini *et al.*, 2016; Masci *et al.*, 2015; Monti *et al.*, 2017; Veličković *et al.*, 2018), which are restrained to a unique fixed graph structure. Our method ameliorates graph spectral approaches by exploiting transfers of spectral bases. Furthermore, our experiments also used a multi-centric, multi-data and publicly available dataset. This provides an exhaustive, reproducible, evaluation for directly exploiting spectral features.

While the potential of our method is demonstrated on cortical parcellation, it can be applied to other analyses of surface data. For instance, our framework has a direct impact on regression problems that involve predictions of cortical thickness over time, potentially leading to new families of geometry-based biomarkers for neurological disorders.

CHAPTER 3

LEARNABLE POOLING IN GRAPH CONVOLUTIONAL NETWORKS FOR BRAIN SURFACE ANALYSIS

Karthik Gopinath[a], Christian Desrosiers[b], Herve Lombaert[c]

[a,b,c] Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

Paper published in *IEEE Transactions on Pattern Analysis and Machine Intelligence* (T-PAMI),
October 2020

Presentation

This chapter presents the article *"Learnable Pooling in Graph Convolutional Networks for Brain Surface Analysis"* (Gopinath, Desrosiers & Lombaert, 2020b) submitted to **TPAMI** (IEEE Transactions on Pattern Analysis and Machine Intelligence), published on 02 October 2020. An initial article was published in the **IPMI** conference (International Conference on Information Processing in Medical Imaging), 2019, presented as an oral talk in Hong Kong. The objective of this article is to develop an adaptive pooling technique for intrinsic aggregation of graph nodes based on graph spectral embedding.

3.1 Introduction

Brain surface analysis plays a crucial role in understanding the mechanisms of perception and cognition in humans (Arbabshirani *et al.*, 2017). However, the complex geometry of the brain surface, comprised of intricate folding patterns, poses considerable challenges in neuroscience. Notably, brain imaging data, for instance acquired by magnetic resonance imaging, typically comes in 3D, a *Euclidean* space, while its analysis often focuses on the thin surface of the brain, a *non-Euclidean* space. This fundamental difference between the domains of acquisition and analysis, coupled with the geometrical complexity of brain surfaces, severely hinders computational approaches for brain surface analysis. As an illustration, neighboring 3D voxels in a neuroimage may in fact represent points that are far apart on the brain surface, as shown on Fig. 3.1. To alleviate this problem, popular surface-based methods (Fischl *et al.*, 2004b; Yeo *et al.*, 2010) often simplify the geometry of the brain, for instance, by mapping the surface to a sphere. This process is, however, computationally expensive. For example, the widely-used surface analysis pipeline of FreeSurfer (Fischl *et al.*, 2004b) requires several hours to inflate the cortical surface to a sphere, match it to an atlas and finally perform a cortical analysis. The geometry of brain surfaces similarly complicates other conventional approaches for brain analysis, such as those based on diffeomorphic transformations (Glaunes, Trouve & Younes, 2004) or on spherical harmonics (Styner *et al.*, 2006).

A key application of brain surface analysis is detecting and tracking the progress of neurodegenerative disorders, such as Alzheimer's disease, which often result in a severe atrophy of brain tissues. Analyzing the geometrical changes of the brain can thus aid in the early diagnosis of such conditions. Initial work has focused on *Euclidean* 3D data based for instance on the texture of magnetic resonance images (Freeborough & Fox, 1998; Vemuri, Gunter, Senjem, Whitwell, Kantarci, Knopman, Boeve, Petersen & Jack Jr, 2008), in order to differentiate Alzheimer's disease from normal aging. While volumetric approaches have shown usefulness in detecting global changes in a Euclidean space (Arbabshirani *et al.*, 2017), surface-based methods (Fischl *et al.*, 2004b; Glaunes *et al.*, 2004; Styner *et al.*, 2006; Yeo *et al.*, 2010) are more adequate for analyzing data on brain surfaces. For example, the analysis of shape abnormalities on brain surfaces has improved the prediction of Alzheimer's disease (Tang, Holland, Dale, Younes, Miller & Initiative, 2014) or the identification of stages in this progressive disorder (Oliveira Jr, Nitrini, Busatto, Buchpiguel, Sato & Amaro Jr, 2010). Nevertheless, all these studies has focused on pre-established measurements of brain surface information. In this paper, we propose to learn and exploit the organizational structure of surface data in order to improve prediction tasks that use data on highly-complex surfaces.

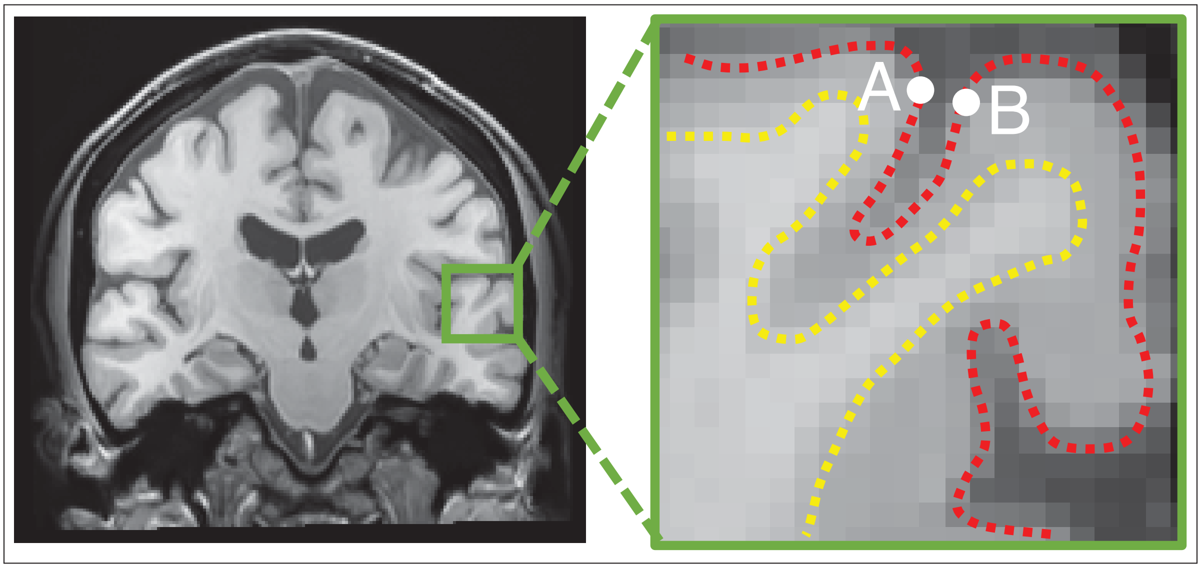


Figure 3.1 Complex geometry of the cerebral cortex – As illustrated, two nearby points in the volume may in fact be far apart on the cortical surface.

3.1.1 Related work

Current machine learning approaches have achieved state-of-the-art performance in a broad range of computer vision and medical imaging applications. In particular, deep learning architectures such as convolutional neural networks (CNNs) (Lecun *et al.*, 1998) offer higher accuracy and speed over traditional approaches for image analysis. In neuroimaging, CNNs are now widely used for various segmentation (Ronneberger *et al.*, 2015) and classification (Wachinger *et al.*, 2017) problems, with architectures tailored for the target task and the available imaging data. For example, various architectures have been proposed to exploit volumetric data (Dolz *et al.*, 2017; Hua *et al.*, 2013; Kamnitsas *et al.*, 2017b; Zhang & Davatzikos, 2011). A fundamental limitation of these models, however, is their restriction to data lying on a fixed Euclidean grid representing pixels or voxels. This restricted representation induces ambiguity when exploiting complex geometries such as in brain surfaces, impeding the application of these Euclidean models for brain surface analysis.

Geometric deep learning (Bronstein *et al.*, 2017) generalizes deep learning models to operate on non-Euclidean domains such as graphs and manifolds. Recent advances in this field, particularly in graph convolutional networks (GCNs), have enabled convolution operations over graphs by exploiting spectral analysis, where convolutions translate into multiplications in a Fourier space (Bruna, Zaremba, Szlam & LeCun, 2014a; Defferrard *et al.*, 2016; Kipf & Welling, 2017; Monti *et al.*, 2017). In such models, convolutions are manipulated with eigenfunctions of graph Laplacian operators (Xu, Fan, Xu, Zeng & Qia, 2018), which can be approximated with Chebyshev (Defferrard *et al.*, 2016) or Cayley polynomials (Levie, Monti, Bresson & Bronstein, 2018). These learned convolution filters can be expressed in terms of mixtures of Gaussians (Monti *et al.*, 2017) or splines (Fey, Lenssen, Weichert & Müller, 2018a). Despite their advantages over standard CNNs, these models are, however, limited to a fixed graph structure and thereby not suitable for brain imaging applications involving a population of subjects. Indeed, brain surfaces have varying geometries with a different number of nodes and a distinct connectivity across meshes. This variability poses computational challenges, for example, arising from the fact that the values of a Laplacian eigenfunction can drastically differ between brains with distinct surface geometries (Ovsjanikov *et al.*, 2012). To this effect, a learned synchronization can correct for differences in eigenfunctions (Yi, Su, Guo & Guibas, 2017). An alignment of eigenbases (Lombaert *et al.*, 2015a) similarly provides a common parameterization of brain surfaces. Such aligned eigenbases enabled the direct learning of surface data across multiple brain geometries (Gopinath, Desrosiers & Lombaert, 2019b). Nevertheless, these types of GCNs are limited to a *fixed* graph structure, for instance, with the same number of nodes.

Standard pooling strategies rely in fact on such consistency of graph structures. Currently, heuristics are often used to mimic a max-pooling strategy in GCNs (Bruna *et al.*, 2014a; Defferrard *et al.*, 2016; Dhillon, Guan & Kulis, 2007). They include varying the number of feature dimensions across layers (Bruna *et al.*, 2014a) while retaining fixed layer sizes, or relying on partition methods, for instance, based binary trees (Defferrard *et al.*, 2016) or Graclus clustering (Dhillon *et al.*, 2007) to coarsen the initial graph. However, these strategies are mainly used for point-wise operations in fixed-size graphs (Monti *et al.*, 2017), such as node classification (Parisot *et al.*, 2017), and do not apply to the task of subject classification when the geometry varies across subjects. A few recent studies (Wang, Samari & Siddiqi, 2018; Ying, You, Morris, Ren, Hamilton & Leskovec, 2018) have attempted to tackle the problem of graph classification in GCNs by incorporating adaptive pooling modules in the network. For instance, (Wang *et al.*, 2018) performs a hierarchical clustering of nodes using their spectral coordinates, with a subsequent pooling of node features within each cluster. While this approach handles varying graph structures, clusters are defined based only on node proximity in the embedding space, without considering node features. Consequently, this unsupervised pooling strategy may not be optimal for the classification or regression task at hand. More recently, a differential pooling technique (Ying *et al.*, 2018) splits the network in two separate paths, one for computing latent features for each node of the input graph and another for predicting the node clusters by which features are aggregated. Similarly, (Gao & Ji, 2019) proposes to use a top-k graph pooling layer in order to down-sample the input graph. This method selects the top-k nodes for the downsampled graph based on a learned projection vector. However, these approaches ignore the intrinsic localization of nodes within the graph, which is sought when the geometry is highly curved such as in brain surfaces.

3.1.2 Contributions

This paper proposes a novel method based on GCNs for classification and regression of surface graphs. Our method includes a learnable pooling strategy which predicts optimal node clusters for each input graph, and thus can handle graphs with varying number of nodes or connectivity. This adaptive pooling technique is applied recursively to obtain a fixed-size representation, which is then used for predicting a target classification or regression value. Our method also leverages spectral embedding techniques for surface graphs (Lombaert *et al.*, 2015a), offering a more powerful representation of complex surfaces like the brain cortex. This contrasts with

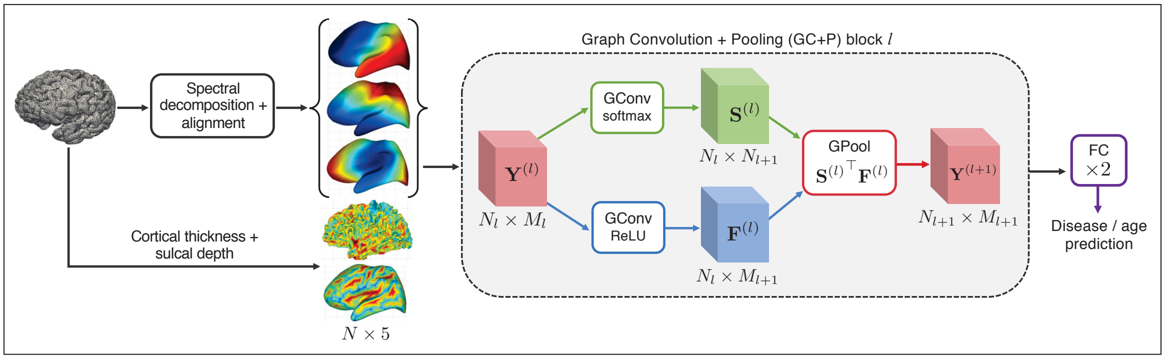


Figure 3.2 An overview of the proposed graph convolutional network – The brain surface graph are mapped to a low-dimensional subspace using spectral decomposition. The spectral bases of the input brain are then aligned to a common reference. Aligned spectral coordinates and cortical surface features are fed as input to the network, composed of sequential Graph Convolution + Pooling (GC+P) blocks and two fully-connected (FC) layers. Each GC+P block processes input node features $\mathbf{Y}^{(l)}$ in two separate paths based on geometric convolutions, one (bottom) deriving a new set of features for each graph node $\mathbf{F}^{(l)}$ and the other (top) computing a soft assignment $\mathbf{S}^{(l)}$ of nodes to clusters representing nodes of the reduced output graph. A pooling layer then obtains reduced graph features $\mathbf{Y}^{(l+1)}$ by aggregating $\mathbf{F}^{(l)}$ in each predicted cluster of $\mathbf{S}^{(l)}$.

the differential pooling approach in (Ying *et al.*, 2018) or (Gao & Ji, 2019), where nodes lack intrinsic localization within the graph.

We illustrate our approach on the challenging tasks of brain surface classification and regression using the well-known Mindboggle (Klein *et al.*, 2017) and ADNI datasets (Jack et al., 2008). We first consider the problem of subject-sex[2] classification and evaluate the impact of our learnable pooling method's hyper-parameters, including the type of pseudo-coordinates, number of clusters, number of eigenvectors, number of neighbors, graph convolution kernel, and input graph size. In an ablation study, we also assess the importance of alignment and regularization for this prediction task. To evaluate the usefulness of our learnable pooling strategy, we compare it against recently-proposed pooling techniques for GCNs.

---

[2] As in most studies, we use the term *sex* instead of *gender* to designate biological differences between male and female subjects.

We show the ability of our pooling strategy to learn important node clusters in a supervised manner by comparing the relationship between these clusters and prominent anatomical regions. To further validate the regions learned by our network, we use it to predict the size of cortical regions as defined by a standard parcellation atlas. Our model is also tested on cortical surface data from the ADNI dataset to (*i*) discriminate between control subjects and subjects suffering from different stages of Alzheimer's, and (*ii*) regress the brain age of subjects. We choose the ADNI dataset (Jack *et al.*, 2008) as it provides manual labels of the subject age and three stages of Alzheimer's disease. Our method achieves a similar performance to the state-of-the-art on the ADNI dataset (Jack *et al.*, 2008), while using only simple cortical measurements such as thickness and sulcal depth.

In summary, the major contributions of our work are as follows:

- A general model for classifying and regressing graphs with varying geometry, which combines a learnable, supervised pooling strategy with the intrinsic (non-Euclidean) localization of nodes via graph spectral embedding.
- A first fully-learned model for brain surface analysis contrasting with previous approaches based on predefined cortical features;
- An in-depth experimental evaluation on two large-scale benchmark datasets (i.e., Mindboggle and ADNI) and four different prediction tasks (i.e., subject-sex classification, cortical region size regression, Alzheimer's disease classification, and brain age regression). Our extensive experiments evaluate the impact of the main components and hyper-parameters of our learnable pooling method, and compares our method against four recently-proposed pooling strategies for GCN;
- State-of-the-art performance for ADNI stages classification and brain age prediction using cortical surface data.

This paper represents a significant extension of our previous work in (Gopinath, Desrosiers & Lombaert, 2019a). Beyond giving a deeper motivation of our work and a more detailed description of the methodology, we thoroughly evaluate our method on a large multi-site dataset, i.e. Mindboggle, as well as on two additional prediction tasks, i.e. subject-sex classification and cortical region-size regression. Added experiments also provide a more comprehensive study of the main hyper-parameters and components of our pooling method and demonstrate its advantage over state-of-art graph pooling techniques relying on unsupervised spectral clustering (Wang *et al.*, 2018), differentiable pooling approaches in Euclidean space (Ying *et al.*, 2018) and a recent top-k pooling method (Gao & Ji, 2019). Moreover, results of new experiments highlight the relationship between the learned clusters for these tasks and known cortical regions, and show the robustness of our method to surface mesh variability in terms of number of nodes and connectivity.

3.2 Method

We first describe a general formulation that extends standard convolutions to non-rigid geometries, such as surfaces. We then detail our strategy based on graph spectral embedding to model the intrinsic localization of mesh nodes and align them across multiple surfaces. Subsequently, we present our end-to-end learnable pooling strategy for the adaptive clustering of graph nodes. Finally, we provide detailed information on the overall network architecture and training procedure.

3.2.1 Convolutions on non-rigid geometries

In a standard CNN, the input is typically provided as a set of features observed over a regular grid of points like 2D pixels or 3D voxels. This information is then processed using a sequence of layers composed of a convolution operation followed by a non-linear activation function like the ReLU. Let $\mathbf{Y}^{(l)} \in \mathbb{R}^{N_l \times M_l}$ be the input feature map at convolution layer l, such that $y_{iq}^{(l)}$ is the q-th

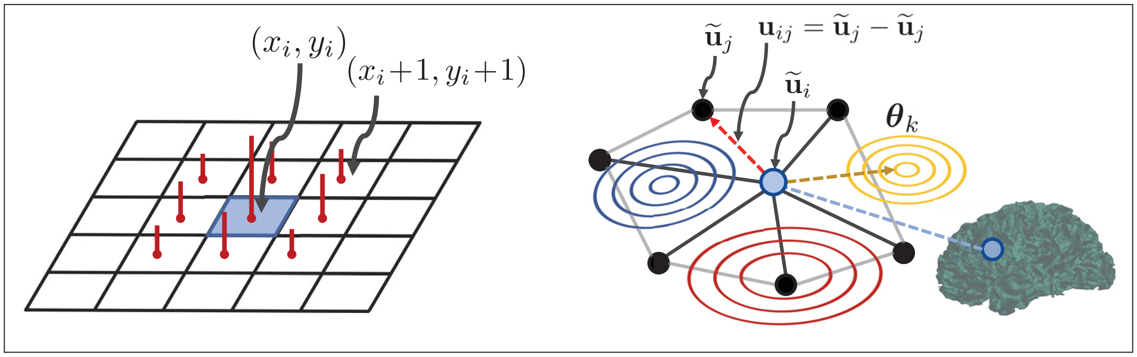


Figure 3.3 Convolutions on grids vs. graphs – Illustration of standard grid-based 2D convolutions (left) and geometric graph convolution (right). The challenge is to exploit kernels on arbitrary graph structures, and to add pooling operations over convolutional layers of graph nodes.

feature of the i-th input node. The feature map consists of N_l input nodes with M_l dimensions each. Assuming a 1D grid for simplicity, the output of layer l obtained by a convolution kernel of size K_l is given by $y_{ip}^{(l+1)} = f(z_{ip}^{(l)})$, where

$$z_{ip}^{(l)} = \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)} \cdot y_{i+k,q}^{(l)} + b_p^{(l)}. \tag{3.1}$$

Here, $w_{pqk}^{(l)}$ are the convolution kernel weights, $b_p^{(l)}$ the weights of the layer, and f the activation function.

For a general surface, points are not necessarily defined on a regular grid and can lie anywhere in a 3D Euclidean space. Such surface can conveniently be represented as a mesh graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V}$ is the set of nodes corresponding to points and $\mathcal{E}$ is the set of edges between the graph nodes. Given a node $i \in \mathcal{V}$, we denote as $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$ the set of nodes connected to i, called neighbors. We extend the concept of convolution to arbitrary graphs using the more general definition of geometric convolution (Fey *et al.*, 2018a; Gopinath *et al.*, 2019b; Monti *et al.*, 2017):

$$z_{ip}^{(l)} \ = \ \sum_{j \in \mathcal{N}_i} \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)} \cdot y_{jq}^{(l)} \cdot \varphi_{ij}(\mathbf{\Theta}_k^{(l)}) \ + \ b_p^{(l)}, \tag{3.2}$$

In this extended formulation, φ_{ij} is a symmetric kernel parameterized by $\mathbf{\Theta}_k$, which encodes the relative position of neighbor nodes j to a node i when computing the convolution at node i. The pseudo-coordinates $\mathbf{u}_{ij}$ of i relative to j are usually defined based on Cartesian or polar coordinates. In this work, we explore two types of kernels for geometric convolutions: the Gaussian kernel (Monti *et al.*, 2017) and B-spline kernel (Fey *et al.*, 2018a). The Gaussian kernel, which has learnable parameters $\mathbf{\Theta}_k = \{\boldsymbol{\mu}_k, \mathbf{\Sigma}_k\}$ corresponding to a mean vector and covariance matrix, computes the response as

$$\varphi_{ij}(\mathbf{\Theta}_k) \ = \ \exp\big(-\tfrac{1}{2}(\mathbf{u}_{ij} - \boldsymbol{\mu}_k)^\top \mathbf{\Sigma}_k^{-1}(\mathbf{u}_{ij} - \boldsymbol{\mu}_k)\big). \tag{3.3}$$

As shown in Fig. 3.3, standard convolutions (left) can be seen as a special case of geometric convolutions with Gaussian kernels (right) where nodes are placed on a regular grid and kernels are unit impulses (i.e., spherical Gaussian kernels with zero variance) placed at the grid position of neighbor nodes. On the other hand, B-spline kernels obtain the response as the product of M_l B-spline basis functions of degree m based on uniform knot vectors. Compared to Gaussian kernels, this kernel has the advantage of making computation time independent from the kernel size, thereby improving computational efficiency and scalability.

3.2.2 Spectral embedding of multiple surface graphs

A significant limitation of the above geometric convolutional model is its inability to process differently-aligned surfaces. Thus, since local coordinates $\mathbf{u}_{ij}$ are determined using a fixed coordinate system, any rotation or scaling of the surface mesh will produce a different response for a given set of kernels. Moreover, as shown in Fig. 3.1, geometric convolutions in Euclidean space are poorly-suited for complex surfaces like the highly-convoluted brain cortex.

We address these issues using a graph spectral embedding approach. Specifically, we map a surface graph $\mathcal{G}$ to a low-dimensional subspace using the eigencomponents of its normalized Laplacian $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{A}$ is the weighted adjacency matrix and $\mathbf{D}$ is the diagonal degree matrix with $d_{ii} = \sum_j a_{ij}$. Although binary adjacency values could be used in $\mathbf{A}$, we instead define the weight between two adjacent nodes as the inverse of their Euclidean distance: $a_{ij} = \big(\|\mathbf{x}_i - \mathbf{x}_j\|_2 + \epsilon\big)^{-1}$ where ϵ is a small constant to avoid a zero-division. Denoting as $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ the eigendecomposition of $\mathbf{L}$, where $\mathbf{\Lambda}$ is the diagonal matrix of real, non-negative eigenvalues, we then compute the normalized spectral coordinates of nodes as the rows of matrix $\widehat{\mathbf{U}} = \mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}$. Here, normalized components are scaled proportionally to the *inverse* of their eigenvalues since components with smaller eigenvalues encode more relevant characteristics of the embedded graph (Chung, 1997). Based on the same principle and as in (Lombaert *et al.*, 2015b), we limit the decomposition to the $d = 3$ first smallest non-zero eigenvalues of $\mathbf{L}$. This allows capturing the important variability of surfaces, while also limiting computational complexity.

We must align the spectral projection of different surface graphs to a common reference $\widehat{\mathbf{U}}^{(0)}$ because the spectral embedding of $\mathbf{L}$ is only defined up to an orthogonal transformation (i.e., rotation or flip). The spectral embedding of a random brain surface in the dataset is chosen as the common reference $\widehat{\mathbf{U}}^{(0)}$. To perform alignment, we find a node correspondence by using an iterative closest point (ICP) approach (Lombaert *et al.*, 2015a), where each node $i \in \mathcal{V}$ is mapped to its nearest reference node $\pi(i) \in \mathcal{V}^{(0)}$ in the embedding space. Denoting as $\widehat{\mathbf{u}}_i$ the normalized spectral coordinates of node i, the alignment task can be expressed as

$$\underset{\pi,\mathbf{R}}{\arg\min} \sum_{i=1}^{N} \big\|\widehat{\mathbf{u}}_i\,\mathbf{R} - \widehat{\mathbf{u}}^{(0)}_{\pi(i)}\big\|_2^2. \qquad (3.4)$$

Let $\widehat{\mathbf{U}}_{\pi}^{(0)}$ be the matrix whose i-th row is $\widehat{\mathbf{u}}_{\pi(i)}^{(0)}$. The transformation between corresponding nodes is approximated as

$$\mathbf{R} \;=\; \left(\widehat{\mathbf{U}}^{\top}\widehat{\mathbf{U}}\right)^{-1}\widehat{\mathbf{U}}^{\top}\widehat{\mathbf{U}}_{\pi}^{(0)} \;=\; \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}^{\top}\widehat{\mathbf{U}}_{\pi}^{(0)}. \tag{3.5}$$

We use the aligned spectral embedding $\widetilde{\mathbf{U}} = \widehat{\mathbf{U}}\,\mathbf{R}$ to define the local coordinates corresponding to an edge $(i, j) \in \mathcal{E}$: $\mathbf{u}_{ij} = \widetilde{\mathbf{u}}_j - \widetilde{\mathbf{u}}_i$. As illustrated in Fig. 3.3 (right), and based on Eq. (3.2), the convolution at node i therefore considers kernel responses $\varphi_{ij}(\mathbf{\Theta}_k^{(l)})$ for neighbor nodes j, *relative to* the spectral coordinates of i.

3.2.3 Learnable pooling for graph convolutional networks

Pooling in standard CNNs is typically carried out by aggregating values inside non-overlapping regions of features maps. In graph convolutional networks (Bruna *et al.*, 2014a; Defferrard *et al.*, 2016; Kipf & Welling, 2017; Monti *et al.*, 2017), however, this approach is not applicable for the following reasons. First, nodes are not laid out on a regular grid, which prevents aggregation of features in predefined regions. Second, the density of points may spatially vary in the embedding space; hence regions of fixed size or shape are not suitable for graphs with different geometries. Last, and more importantly, input surface graphs may have a different number of nodes, while the output may have a fixed size. This is the case when predicting a fixed number of class probabilities from different brain geometries.

We propose an end-to-end learnable pooling strategy for the subject-specific aggregation of cortical features, inspired by the differential pooling technique of Ying et al. (Ying *et al.*, 2018). Our strategy, shown in Fig. 3.2, produces a sequence of convolutional feature maps $\{\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(l)}, \ldots, \mathbf{Y}^{(L)}\}$, with $\mathbf{Y}^{(l)} \in \mathbb{R}^{N_l \times M_l}$, by the repeated application of a Graph Convolution + Pooling (GC+P) block. Each GC+P block takes as input a feature map $\mathbf{Y}^{(l)}$ on a N_l node graph, and processes it in two separate paths: the first one computing latent features for each node of the input graph and the second predicting the node clusters by which the features are aggregated. The feature encoding path applies a sequence of geometric convolutions as in Eq. (3.2) to generate a new feature map $\mathbf{F}^{(l)} \in \mathbb{R}^{N_l \times M_{l+1}}$ on the block's input graph. The clustering path also consists of sequential geometric convolutions, however the activation function of the last convolution is replaced by a node-wise softmax. The output of this last convolution, $\mathbf{S}^{(l)} \in [0, 1]^{N_l \times N_{l+1}}$, gives for each node i the probability s_{ic} that i belongs to cluster $c \in \{1, \ldots, N_{l+1}\}$.

Pooled features $\mathbf{Y}^{(l+1)} \in \mathbb{R}^{N_{l+1} \times M_l}$ are computed as the expected sum of convolutional features in each cluster c, i.e.

$$\begin{aligned} y_{cp}^{(l+1)} &= \sum_{i=1}^{N_l} s_{ic}^{(l)} \cdot f_{ip}^{(l)} \\ \mathbf{Y}^{(l+1)} &= \mathbf{S}^{(l)\top} \mathbf{F}^{(l)}. \end{aligned} \tag{3.6}$$

The processing of aggregated node features, downstream the pooling operation, requires computing a new adjacency matrix $\mathbf{A}^{(l+1)}$ and spectral coordinates $\widetilde{\mathbf{U}}^{(l+1)}$ for the node clusters which become the nodes of the block's reduced-size output graph. Here, we define the adjacency weights between pooling clusters c and d as

$$\begin{aligned} a_{cd}^{(l+1)} &= \sum_{i=1}^{N_l} \sum_{j=1}^{N_l} s_{ic}^{(l)} \cdot s_{jd}^{(l)} \cdot a_{ij}^{(l)} \\ \mathbf{A}^{(l+1)} &= \mathbf{S}^{(l)\top} \mathbf{A}^{(l)} \mathbf{S}^{(l)}. \end{aligned} \tag{3.7}$$

Intuitively, $a_{cd}^{(l+1)}$ is the expected number of connected nodes between clusters c and d. Likewise, the spectral coordinates of cluster c is computed as the mean coordinates (i.e., centroid) of all nodes assigned to c:

$$\begin{aligned} \widetilde{u}_{cp}^{(l+1)} &= \sum_{i=1}^{N_l} s_{ic}^{(l)} \cdot \widetilde{u}_{ip}^{(l)} \\ \widetilde{\mathbf{U}}^{(l+1)} &= \mathbf{S}^{(l)\top} \widetilde{\mathbf{U}}^{(l)}. \end{aligned} \tag{3.8}$$

The bilinear formulation of Eq. (3.6) faces a challenging optimization problem with several local minima. For instance, the same output $\mathbf{Y}^{(l+1)}$ in Eq (3.6) can be obtained by modifying either $\mathbf{S}^{(l)}$ or $\mathbf{F}^{(l)}$. To alleviate this problem and obtain spatially-smooth clusters, we add a Laplacian regularization term to the loss function:

$$\begin{aligned} \mathcal{L}_{\text{reg}}(\mathbf{S}^{(l)}) &= \sum_{i=1}^{N_l}\sum_{j=1}^{N_l} a_{ij}^{(l)} \cdot \|\mathbf{s}_i^{(l)} - \mathbf{s}_j^{(l)}\|_2^2 \\ &= \operatorname{tr}\big(\mathbf{S}^{(l)}\mathbf{L}^{(l)}\mathbf{S}^{(l)\top}\big), \end{aligned} \tag{3.9}$$

where $\mathbf{s}_i^{(l)}$ denotes the cluster probability vector of node i (i.e., the i-th row of $\mathbf{S}^{(l)}$). This well-known regularization approach (Belkin, Niyogi & Sindhwani, 2006) penalizes connected nodes to be mapped to different clusters, with a penalty proportional to the connection strength.

3.2.4 Architecture details

Figure 3.2 presents the overall architecture of our graph convolutional network. As input, we give to the network the cortical surface features $\mathbf{x}_i$ and aligned spectral coordinates $\widetilde{\mathbf{u}}_i$ of each node i. For computing graph convolutions as in Eq. (3.2), we define the neighbors $\mathcal{N}_i$ of node i as the $k = 5$ nodes nearest to i in the spectral embedding (i.e., the distance between node i and j corresponds to $\|\widetilde{\mathbf{u}}_i - \widetilde{\mathbf{u}}_j\|_2$) plus node i itself. While various features could be considered to model the local geometry of the cortical surface (Fischl *et al.*, 2004b), we considered sulcal depth and cortical thickness in this work, since the first one helps delineate anatomical brain regions (Destrieux *et al.*, 2009) and the latter is related to ageing (Sowell et al., 2004) and neurodegenerative diseases such as Alzheimer's (Lerch et al., 2004).

The network comprises two cascaded GC+P blocks, followed by two fully-connected (FC) layers. The first block generates an N×8 feature map and an N×16 cluster assignment matrix, in two separate paths, and combines them using the pooling formulation of Eq. (3.6) to obtain a pooled feature map of 16×8. In the second block, pooled features are used to produce a 16×16 map of features, pooled in a single cluster. Hence, the second pooling step acts as an attention module selecting the features of most relevant clusters. The resulting 1×16 representation is converted to a 1×8 vector using the first FC layer, and then to a 1× *NumOutputs* vector with the second FC layer, where the number of outputs *NumOutputs* depends on the prediction task.

Except for the cluster probabilities and network output, all layers employ the Leaky ReLU (Nair & Hinton, 2010) as activation function: $y_{ip}^{(l)} = \max(0.01 z_{ip}^{(l)}, z_{ip}^{(l)})$. In the default setting of our pooling method, for the graph convolution kernel φ_{ij} of Eq. (3.2), we used the B-spline kernel proposed by Fey et al. (Fey *et al.*, 2018a). However, we also test the Gaussian kernel (Monti *et al.*, 2017) in our experiments.

For training, the loss function combines the output prediction loss and cluster regularization loss on the first GC+P block:

$$\mathcal{L}(\mathbf{\Theta}) = \mathcal{L}_{\text{out}}(\mathbf{\Theta}) + \alpha \mathcal{L}_{\text{reg}}\big(\mathbf{S}^{(1)}\big), \tag{3.10}$$

where α is a parameter controlling the amount of regularization. For classification tasks (i.e., disease prediction), $\mathcal{L}_{\text{out}}$ is set as the cross-entropy between one-hot encoded ground-truth labels and output class probabilities. In the case of regression (i.e., brain age prediction), we use mean squared error (MSE) for this loss. Network parameters are optimized with stochastic gradient descent (SGD) using the Adam optimizer. Experiments were carried out on an i7 desktop computer with 16GB of RAM and an Nvidia Titan X GPU. The model takes less than a second for disease classification or age regression.

3.3 Experiments and results

We validate our method on two large-scale, publicly-available datasets: Mindboggle-101 (Klein *et al.*, 2017) and ADNI1 (Jack *et al.*, 2008). The first one contains T1-weighted MRI from 101 healthy subjects (males: n=57, females: n=44, age: 20–61 years) collected from 9 different sites. We use this dataset for the tasks of subject-sex classification and cortical region size regression since both subject-sex labels and manual annotations for 32 cortical parcels are provided with imaging data. The ADNI1 dataset (Jack *et al.*, 2008) is comprised of multi-sequence MRI data

Table 3.1 Impact of our hyper-parameters on our learnable pooling method – Mean and standard deviation were computed on 5 separate runs using a different random 50K node sub-sampling of each graph. For every hyper-parameter, the default setting of our method is highlighted in bold font.

Experiments	**Parameters**	**Mean** ± **Std.**
Pseudo-coordinates	Cartesian	80.40 ± 4.21
	Polar	83.15 ± 2.10
	Ours - Spectral	**84.21** ± **3.72**
Number of clusters	4	73.68 ± 5.76
	8	76.84 ± 7.87
	16	**84.21** ± **3.72**
	32	77.89 ± 2.10
Number of eigenvectors	Only cortical features	70.52 ± 5.36
	1	75.78 ± 7.13
	3	**84.21** ± **3.72**
	5	77.89 ± 2.10
	10	74.73 ± 8.40
Number of neighbors	2	81.05 ± 2.57
	3	82.10 ± 2.57
	5	**84.21** ± **3.72**
	10	84.21 ± 3.93
Graph convolution kernel	Gaussian (Monti *et al.*, 2017)	83.15 ± 2.15
	B-Spline (Fey *et al.*, 2018a)	**84.21** ± **3.72**
Ablation study	W/o Alignment	69.47 ± 8.42
	W/o Regularization	74.73 ± 5.15

from 400 subjects diagnosed with mild cognitive impairment (MCI), 200 subjects with early Alzheimer's disease (AD) and 200 elderly control subjects (NC), obtained from 55 participating sites. Both datasets contain brain surface meshes with pointwise cortical thickness and sulcal depth measurements, generated by FreeSurfer (Fischl *et al.*, 1999). Cortical meshes in these datasets vary from 102K to 185K nodes. The code for our work is available at the following URL: https://github.com/kharitz/learnpool.git.

In the first series of experiments, we evaluate the effects of hyper-parameters influencing the performance of our pooling method. Next, an ablation study is presented to assess the effect of our spectral alignment and our Laplacian regularization. Different pooling strategies for our graph convolutional network are thereafter compared on the subject-sex classification problem, while also evaluating the impact of input graph size on prediction accuracy. We then illustrate our network's ability to learn meaningful node clusters by predicting the size of cortical parcels from an anatomical atlas. Finally, we highlight the advantages of working in the spectral domain on the problems of disease classification (NC *vs* AD, MCI *vs* AD, and NC *vs* MCI) and brain age regression.

3.3.1 Impact of hyper-parameters

Our learnable pooling method requires the selection of several hyper-parameters: the type of pseudo-coordinates, the number of clusters, the number of eigenvectors, the number of neighbors, and the type of graph convolutions. In the next series of experiments, we assess the impact of each of these hyper-parameters on the task of subject-sex classification with the MindBoggle dataset, using a 70-10-20 split for training, validation, and testing. To have a measure of variance, keeping the same split, we generated 5 different subsets by randomly sub-sampling 50K nodes in each training, validation and testing graph, and used the sub-sampled graphs as input to our model. Performance (mean and standard deviation) is measured across 5 runs, each one carried out on a different subset. The same architecture, shown in Fig. 3.2, is used across the following experiments.

3.3.1.1 Pseudo-coordinates

We first evaluate the benefit of using spectral information when computing the pseudo-coordinates of nodes in the graph convolution kernel, by comparing it against conventional Cartesian and polar coordinates. The same architecture of Fig. 3.2, based on B-spline kernels, is used for all three settings. As reported in Table 3.1 accuracy improvements of 3.81% and 1.06% are obtained over Cartesian and polar coordinates, respectively, showing the ability of spectral pseudo-coordinates to better capture the local geometry of a complex surface. Note that, to have a fair comparison, spectral node coordinates were used as input to the network in all three settings, hence the models using Cartesian and polar pseudo-coordinate also leverage spectral information. Comparing Cartesian and polar pseudo-coordinates together, we find that polar ones provide a higher accuracy. While both encode similar information, polar coordinates offer a more direct description of distance and direction between two points, which could help to learn their relation. This may explain why polar pseudo-coordinates were preferred in earlier work (Monti *et al.*, 2017).

3.3.1.2 Number of clusters

Next, we train our GCN network using different numbers of clusters for the pooling operation of the network's first GC+P block. As presented in Table 3.1, four settings are tested: 4, 8, 16 and 32 clusters. We see a regular increase in accuracy from 73.64% to 84.21% when going from 4 to 16 clusters. This reflects the fact that sex-related differences are present in various cortical regions, which can be learned by the network. However, the accuracy drops significantly when further increasing the number of clusters to 32. This could be due to the creation of near-empty clusters that add no useful information to the training while increasing the number of parameters to learn.

3.3.1.3 Number of eigenvectors

The inputs of our GCN are the aligned spectral components (the Laplacian matrix eigenvectors) and two cortical features, i.e., sulcal depth and cortical thickness, corresponding to each mesh node. In the next experiment, we vary the number of spectral components given as input, testing five different settings: 0 (only cortical features), 1, 3, 5, and 10. For all settings, three eigenvectors are used to compute pseudo-coordinates in the graph convolutions, and the same 70-10-20 split as the previous experiments is employed. Results presented in Table 3.1 demonstrate the importance of including spectral information as input, with an accuracy improvement of 5.26% when adding the first component (i.e., eigenvector with smallest non-zero eigenvalue) to cortical features. The best performance of 84.21% is achieved when considering the first three eigenvectors in addition to cortical features. A possible explanation for this result is that three is the minimal number of eigenvectors required to uniquely locate a point on a 3D surface (Lombaert *et al.*, 2015a). As suggested by the decreasing accuracy for 5 and 10 spectral components, higher-order eigenvectors may capture highly-varying and subject-specific patterns of sulcal and gyral geometry, which is not predictive of subject sex.

3.3.1.4 Number of neighbors

The number of neighbors k directly impacts the computation of convolutions in Eq. (3.2). To better assess the effect of this hyper-parameter, the performance of a classification task is evaluated while increasing the number of neighbors within randomly sub-sampled graphs of 50K nodes. More precisely, for every node i in the graph, the k nearest neighbors are defined in the spectral embedding space. The smoothness of the Laplacian matrix eigenvectors ensures that neighbors are locally close to each other on the brain surface. Performance is then evaluated using a classification model that is trained on sub-sampled graphs with $k = 2$, 3, 5, and 10 neighbors.

Table 3.1 shows a higher classification accuracy when the number of neighbors increases. From a classification accuracy of 81.05% for $k = 2$ neighbors, the performance improves gradually to 84.21% with $k = 5$ and $k = 10$ neighbors. However, the computational overhead of employing larger neighborhoods must also be taken into account. For instance, runtime increases by a factor of 1.7, from 93.6 ms to 158.5 ms, when going from $k = 5$ to $k = 10$. For this reason, a neighborhood size of $k = 5$ is used in the default setting of our method.

3.3.1.5 Graph convolution kernel

Last, we compare the B-spline convolution of our default architecture with the Gaussian kernel of (Monti *et al.*, 2017) with diagonal covariance matrix. As reported in Table 3.1, we observe a small improvement of 1.06% when employing B-spline kernels. Interestingly, the number of parameters is almost the same for both kernel types (2,257 for Gaussian compared to 2,233 for B-spline) and, thus, performance differences are not due to overfitting. Note that similar improvements of B-spline kernels compared to Gaussian kernels were also observed in (Fey *et al.*, 2018a) for various analysis tasks.

3.3.2 Ablation study on alignment and regularization

In this section, we perform an ablation study to evaluate the contribution of spectral alignment and Laplacian regularization on our method's performance for the task of predicting the sex of Mindboggle subjects. To assess the usefulness of spectral alignment, we first train a model with unaligned spectral coordinates and with cortical features. The results in Table 3.1 show the unaligned components to yield a low accuracy of 69.47%, demonstrating the importance of this alignment for learning across different surfaces. Results also highlight the role of Laplacian regularization in learning, with a 9.48% drop in accuracy when removing the corresponding term in the loss of Eq. (3.10), i.e. using $\alpha = 0$ in the loss. As explained before, this regularization term is necessary to avoid getting stuck in a local minimum caused by the bi-linear formulation of the pooling operation. Laplacian regularization also provides spatially-smoother clusters that better reflect the underlying anatomy of the brain.

3.3.3 Comparison of different pooling methods

We compared our learnable pooling strategy against four other pooling techniques applicable to graph convolutional networks: 1) taking the global average of feature maps, 2) pooling feature maps in fixed regions computed from a cortical parcel atlas, 3) pooling the same features in

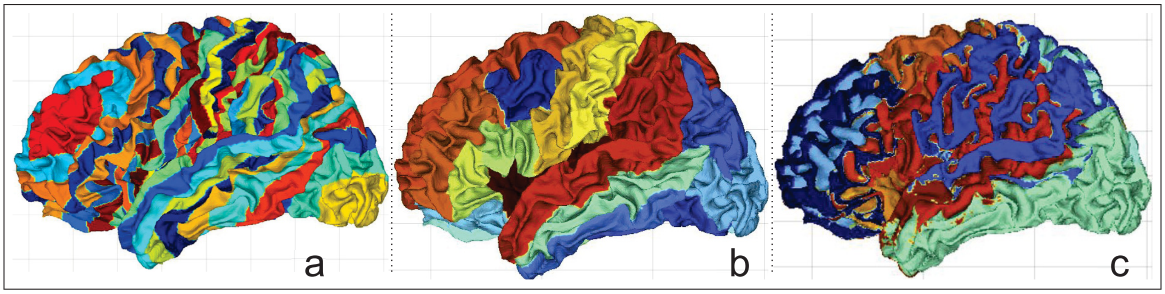


Figure 3.4 Clusters of different pooling methods – (a) Clusters obtained by spectral k-means clustering. (b) Fixed clusters computed from a cortical parcel atlas. (c) Clusters learned by our learnable pooling method. Colors on the brain surface represent different regions.

regions obtained by applying k-means clustering on the spectral embedding, 4) the top-k pooling approach proposed in (Gao & Ji, 2019) for downsampling.

For all tested methods, we used a network composed of two graph convolution layers followed by two fully-connected layers, as described in Section 3.2.4. In the case of global average pooling and fixed parcellation pooling, a single pooling operation is applied after the second graph convolution. For spectral clustering pooling, nodes are grouped after each of the two convolution layers as in our learnable pooling. However, the pooling path of the network is replaced by a static node clustering. Likewise, for top-k pooling, we employ the same architecture as presented in Section 3.2.4, but replace our pooling path with the top-k pooling after the graph convolution operation. We train and test all methods on subject-sex classification using the MindBoggle dataset with a 70-10-20 split for training, validation, and testing. Once again, we perform 5 separate runs with a different random sub-sampling of 50K nodes for each graph.

Table 3.2 summarizes the results of this experiment. We see that global average pooling yields the poorest performance with a mean accuracy of 60.76%. Using atlas-defined cortical parcels to aggregate features improves accuracy slightly to 64.59%, suggesting that these parcels are informative for identifying subject sex. Moreover, applying unsupervised clustering on the spectral embedding further increases mean accuracy to 67.94%, which indicates the benefits of having a hierarchy of non-fixed clusters. The advantage of a learnable top-k pooling over fixed pooling methods can be seen with a classification accuracy of 78.92%. However, by learning clusters in a supervised manner from spectral embeddings, our method achieves the highest accuracy of 81.33%, an improvement of 13.39% over spectral clustering and a 5.3% gain over top-k pooling.

Figure 3.4 gives examples of clusters for the different pooling strategies (except global average pooling, which considers all nodes as part of a single cluster and top-k pooling as it selects nodes to drop for downsampling). While spectral clustering yields spatially-regular clusters, the distribution of these clusters is arbitrary and does not seem to match known parcels of the cortex (shown in Fig. 3.4b). In contrast, the clusters predicted by our pooling strategy are larger and better aligned with these known parcels.

Table 3.2 Baseline graph pooling methods comparison – Mean and standard deviation were computed on 5 separate runs using a different random 50K node sub-sampling of each graph

Pooling method	**Mean ± Std.**
Global Average Pooling	60.76 ± 3.62
Fixed Parcellation Pooling	64.59 ± 7.84
Spectral Clustering Pooling (Wang *et al.*, 2018)	67.94 ± 4.97
Top-k pooling (Gao & Ji, 2019)	78.94 ± 3.32
Learnable Pooling (ours)	**84.21 ± 3.72**

3.3.4 Impact of input graph size

In the next experiment, we investigate whether our method is robust to variability in the size of the surface mesh. Toward this goal, we use the same split of the MindBoggle dataset as in the first experiment, and randomly sub-sample the original mesh to 100, 1K, 5K, 10K, 25K, 50K and 75K nodes. Because convolutions at each node use information from its $k = 5$ nearest neighbors, as described in Eq. (3.2), testing multiple sub-sampling with the same number of nodes also assesses the robustness of our model to variations in graph connectivity. We train our model on each of these reduced graph datasets to predict the sex of MindBoggle subjects.

Table 3.3 gives the classification accuracy for different sizes of training graphs when testing on sub-sampled graphs of the same size, or on the original full-sized graph. The first case evaluates whether the same accuracy can be achieved with less information at the input of the network, whereas the second case tests if the convolution parameters learned by the network generalize to larger graphs. As expected, classification performance decreases when reducing the size of input graphs, both when testing on sub-sampled graphs and full-sized graphs. When testing on sub-sampled graphs, accuracy drops from 94.73% while training with full graphs to 55.02% for graphs with only 100 nodes. However, high accuracy values of 84.21% and 85.26% can be achieved when training graphs of 50K and 75K nodes, respectively, about half the size of the original graphs. Furthermore, we see that our model trained with moderately-reduced graphs can still perform well on full-sized ones. For instance, the model trained with graphs of 50K nodes and 75K nodes achieves an accuracy of 78.94% and 84.21% respectively, when tested on original graphs with twice the number of nodes.

Table 3.3 Subject-sex classification performance of our pooling approach on different sub-graphs – Mean classification accuracy (%) with standard deviation over test set from the Mindboggle dataset

Num. of nodes	**Testing on Sub-sampled graphs**	**Testing on Full graphs**
100	55.02 ± 13.18	52.63 ± –
1k	55.98 ± 4.25	52.63 ± –
5k	64.11 ± 1.58	47.36 ± –
10k	67.94 ± 5.98	52.63 ± –
25k	71.77 ± 4.86	73.68 ± –
50k	84.21 ± 3.72	78.94 ± –
75k	85.26 ± 3.93	84.21 ± –
Full graph	94.73 ± –	94.73 ± –

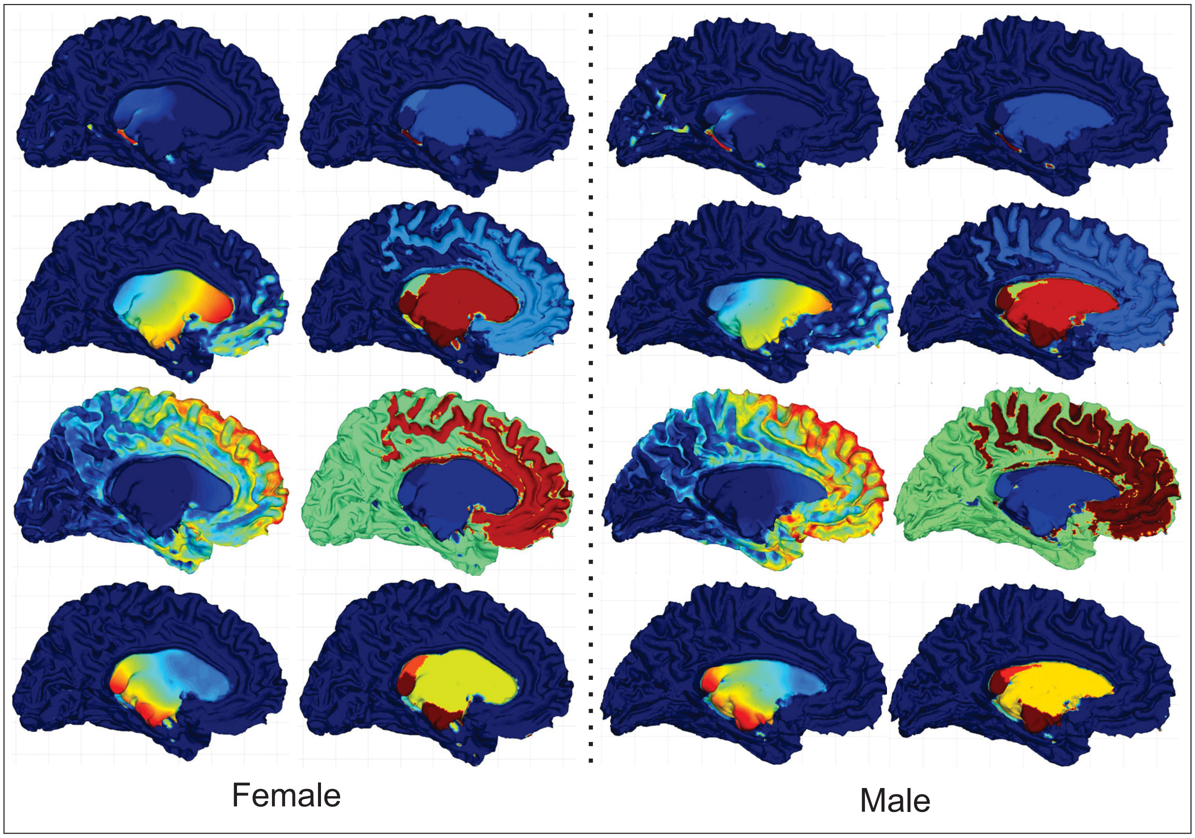


Figure 3.5 Feature maps and predicted clusters for the task of subject-sex classification – The first column shows examples of activation maps computed by the embedding path of our network for a female subject. The second column gives the average activation in each predicted cluster for the same subject and feature maps. Coloring indicates output of the ReLU activation with minimum value indicated by blue and maximum value indicated by red. Third and fourth columns depict the same information for a male subject.

3.3.5 Task-specific pooling regions

In this section, we qualitatively and quantitatively evaluate the predicted clusters and feature maps learned by our network. Once more, we consider the task of classifying males *vs.* females subjects from the Mindboggle dataset with the architecture depicted in Fig. 3.2.

Figure 3.5 shows examples of features and clusters learned by our graph pooling model for a male and a female subject. The first and third columns give the distribution of four different activation maps learned by the network for the two subjects. The mean activation in each predicted cluster for the same subjects is illustrated in the second and fourth columns of the figure. We observe the diversity of depicted clusters, spawning different regions of the brain both on the cortex and around regions of the basal ganglia. Interestingly, several of the learned clusters focus on sub-cortical regions like the hippocampus (first row) and amygdala (last row), which have been linked to sex-related differences in the literature (Murphy, DeCarli, Mclntosh, Daly, Mentis, Pietrini, Szczepanik, Schapiro, Grady, Horwitz et al., 1996). This illustrates the benefit of learning task-specific clusters in a supervised manner. Additionally, we see that predicted feature maps and clusters in both subjects are similar, demonstrating that the model can adapt to the specific brain geometry of individual subjects.

We further evaluate the relevance of learned clusters by training the same model to predict the size of 32 anatomical parcels of each brain surface, using labeled data from Mindboggle. For this experiment, we hypothesize that the network should learn clusters that are related to the predefined parcels. To do so, we modify the last layer of the architecture in Fig. 3.2 to have 32 outputs, one for the size of each parcel, and change the loss function to mean square error. Adjusted mutual information (AMI) is used to measure the similarity between learned clusters and ground-truth parcels. AMI values range from 0 to 1, a score of 0 corresponding to random clusters and a score of 1 for clusters identical to ground-truth.

Figure 3.7 gives the mean AMI obtained at each training epoch, and examples of predicted clusters at four different epochs are shown in Fig. 3.6. In the initial stages of training, the model predicts a small number of clusters corresponding mainly to the components of the spectral embedding (see the network input in Fig. 3.2). In the first 500 epochs, the AMI score between predicted clusters and ground-truth parcels drops. Then, as training progresses, we observe increasing AMI values and progressively more defined clusters. At the end of training (2500 epochs), the model achieves an AMI score of 0.39. Obtained clusters appear to be a combination of different ground-truth parcels, suggesting that fully-connected layers further help regressing parcel sizes.

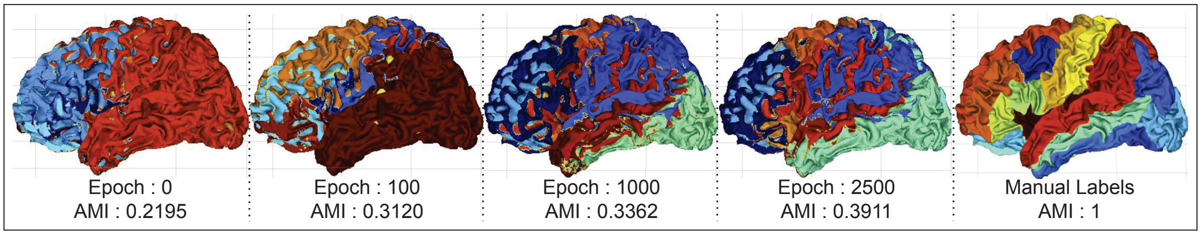


Figure 3.6 Pooling regions learned during training – The pooling regions are learned for the model training to regress the size of cortical regions. During initial epochs, random regions are clustered together to aggregate feature maps. A low AMI score indicates this random clustering compared to the ground-truth. After training, the model finally learns to group multiple parcels (cyan) into on cluster pooling region. AMI score increases over epochs, indicating task-dependent learning by our model. The last figure shows manual parcels with AMI score of 1 for reference.

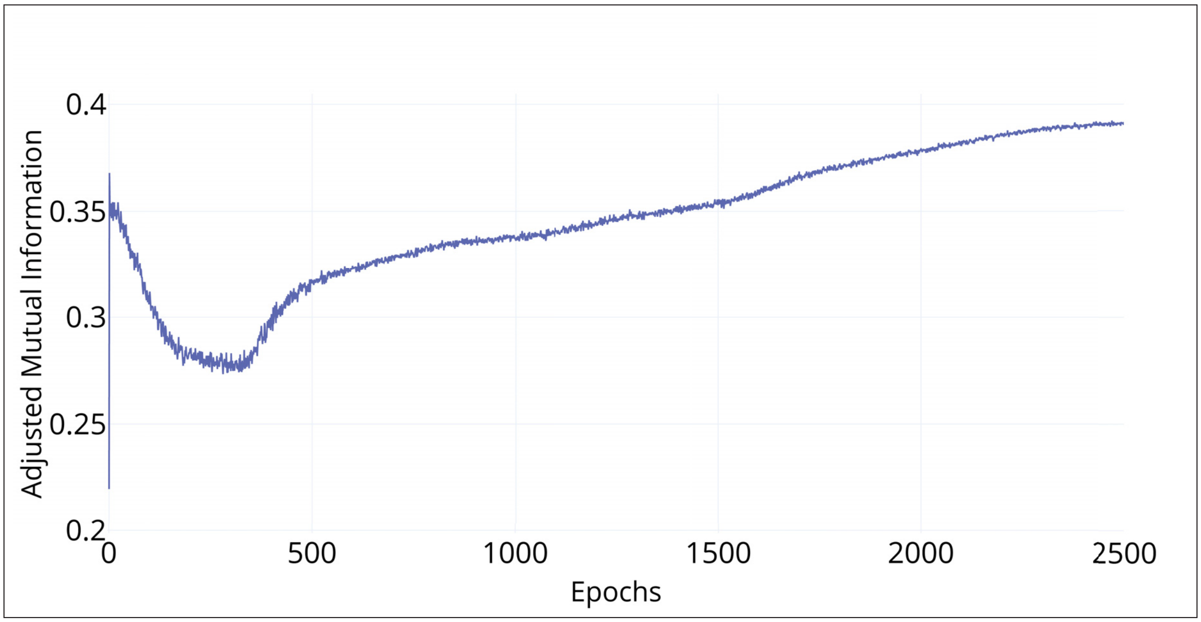


Figure 3.7 Evolution of AMI score – The adjusted mutual information score between the pooling regions and the manual parcels over multiple epochs is shown. A random overlap between learned pooling regions and parcels is observed at initial epochs. After training, the AMI score increases with the pooling regions corresponding to ground-truth parcels.

3.3.6 Disease classification

In the following experiment, we evaluate our method on the task of classifying subjects from the ADNI dataset as normal control (NC), mild cognitive impairment (MCI) or Alzheimer's disease (AD). Specifically, we consider three different binary classification problems: NC *vs* AD, MCI *vs* AD and NC *vs* MCI. We compare our method against the random forest approach in (Ledig *et al.*, 2014), which also considers surface-based information from the ADNI dataset. To measure the contribution of the spectral embedding in our method, we also evaluate our model trained with only cortical thickness and sulcal depth as input. The same random split of 70-10-20 is employed for all three models.

The classification performance of tested models is reported in Table 3.4. We see that our method outperforms the random forest approach of (Ledig *et al.*, 2014) on all three classification problems. Relative to this approach, the proposed method yields mean accuracy improvements between 7.79% and 11.92%. A significant gain in performance is also observed when comparing the same method trained without spectral node coordinates. This is particularly notable for NC *vs* MCI, where adding spectral coordinates increases the mean accuracy by 13.33%. Note that we have also tried giving the network original (x, y, z) coordinates of mesh nodes. However, this led to worse results. This illustrates the advantage of using intrinsic node localization when processing surface data.

Table 3.4 Evaluation of the proposed work – Average accuracy of disease classification (%), with standard deviation over the complete ADNI dataset. First row is a random forest (RF) with multiple cortical-based features (Ledig *et al.*, 2014). Second row is our graph convolutional model without geometrical information (spectral node coordinates $\widetilde{\mathbf{U}}$). Last row is with this information.

Input	**NC *vs* AD**	**MCI *vs* AD**	**NC *vs* MCI**
RF (Ledig *et al.*, 2014)	80 ± 5	65 ± 6	63 ± 4
Ours w/o $\widetilde{\mathbf{U}}$	76.00 ± 6.06	74.03 ± 8.63	63.71 ± 5.72
Ours with $\widetilde{\mathbf{U}}$	89.33 ± 4.30	76.92 ± 4.78	70.79 ± 6.40

3.3.7 Brain age prediction

The last experiment demonstrates our method in a regression problem where the age of NC subjects of the ADNI dataset is predicted using pointwise surface-based measurements. In this

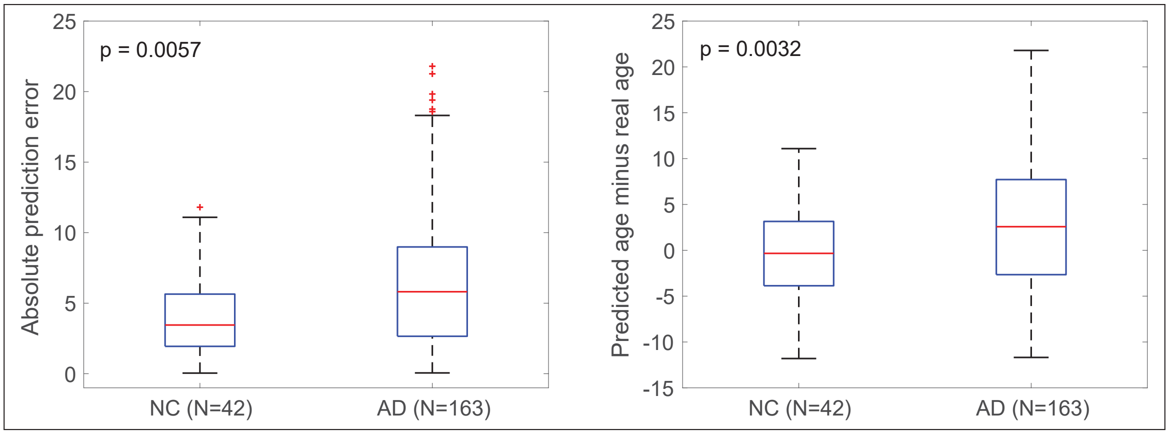


Figure 3.8 Evaluation of brain age prediction – Distribution of absolute prediction error (left) and predicted minus real age (right), for NC and AD test subjects. Our learnable pooling strategy yielded graph models that could correctly capture age discrepancies between real and geometry-based ages, as expected between subjects with NC and AD.

case, the network outputs a single value, and MSE is used as loss function. Once more, we test our method trained with or without spectral node coordinates as input. Moreover, to evaluate brain age prediction as a potential imaging biomarker for Alzheimer's, we also measure the prediction accuracy of our model on AD test subjects.

The results of this experiment are summarized in Fig. 3.8, which gives the distribution of mean absolute error (MAE) and prediction bias (predicted age minus real age) for NC subjects and AD subjects. When testing on NC subjects, our method achieves an MAE of 4.35 ± 3.19 years, which is comparable with results in the literature. As expected, a higher MAE of 6.80 ± 6 years is obtained for AD subjects, since the symptoms of early Alzheimer's are similar to premature brain aging. The brain age, calculated as the predicted age minus the real age, shows a statistically significant difference with a p-value of 0.0032. This value suggests the potential application of brain age prediction as a biomarker for AD.

3.4 Conclusion

We presented a novel strategy that enables pooling operations on arbitrary graph structures. The performance of our learnable pooling scheme was evaluated in seven experiments.

The first series of experiments evaluated the impact of hyper-parameters: the type of pseudo-coordinates of nodes in graph convolution kernels, showing improvement when employing our spectral-based coordinates instead of Cartesian or polar-based ones; the number of clusters in pooling operations, with a regular increase of performance up to 16 clusters; the number of eigenvectors, suggesting that a minimal number of three Laplacian eigenvectors is necessary for optimal accuracy; the number of neighbors, revealing a compromise between accuracy and computation time; the type of graph convolution kernel, showing an improvement of accuracy when using B-spline convolution kernels in our default architecture instead of Gaussian kernels.

A second experiment provided an ablation study validating the positive effects of spectral alignment and Laplacian regularization in our method. Results showed a significant performance gain when using both techniques, compared to employing only one of them.

A third experiment compared different pooling techniques for graph convolutional networks on the subject-sex classification task. A simple global average pooling failed to capture geometric information from consecutive layers, yielding a low performance of 60%. In comparison to employing fixed pooling regions, learning these regions with unsupervised clustering, or applying the top-k pooling strategy to select nodes from a learned projection vector, our learnable pooling strategy offers significantly higher accuracy.

A fourth experiment evaluated the effect of the graph size on the performance of subject-sex classification. The results showed that small graphs lack information to capture the complete geometry of surfaces. However, reducing the size of the graph by 25% up to 75K still yields a high accuracy, while improving memory and computational requirements.

The fifth experiment explored the relationship between learned features and anatomy. The visualization of activation maps and clusters in the network revealed diversity in terms of brain regions. Several learned clusters highlighted essential regions of the basal ganglia, such as the hippocampus and amygdala, which are associated with sex-related differences in the literature. We further evaluated this result with an experiment to regress the size of cortical parcels. As expected, the trained model learns pooling regions similar to the manually-annotated parcels.

The sixth experiment focused on predicted stages of Alzheimer's disease from surface data, including cortical thickness and sulcal depth. Our results showed that pointwise surface values could be efficiently aggregated into a fixed number of class probabilities using the proposed network architecture. Compared to another approach exploiting surface-based features (Ledig *et al.*, 2014), our method achieved significant improvements ranging from 7% to 11%. This performance gain is mainly due to including spectral coordinates of graph nodes as input to the network, demonstrating the importance of intrinsic node localization.

In a final experiment, the age of ADNI subjects was predicted using pointwise surface data. Results showed that our method provides an accuracy comparable to previous approaches in the literature, while using only surface-based information. As expected, subjects with Alzheimer's have higher discrepancies than subjects with normal cognition (Fig. 3.8). The potential of the proposed method as an imaging biomarker for AD could be evaluated in a future study.

To summarize, the proposed pooling strategy enables the exploration of a new family of architectures for graph convolutional networks. Our method exploits the spectral embeddings of graph nodes in order to learn spatially-representative pooling patterns across different layers. However, this requires having datasets of comparable brain geometry, since the eigen-decomposition of the graph Laplacian matrix assumes that shapes are topologically equivalent. Differences in the meshing procedure as well as the presence of holes or cuts in the mesh, for instance caused by ablated tumors, might therefore impact the performance of our method. In future work, we plan to investigate domain adaptation techniques, for example based on adversarial learning, to learn an internal representation which is robust to such differences. Moreover, by incorporating unpooling operations in the proposed model, we could also explore applications requiring node-level outputs like regressing cortical thickness over time.

CHAPTER 4

GRAPH DOMAIN ADAPTATION FOR ALIGNMENT-INVARIANT BRAIN SURFACE SEGMENTATION

Karthik Gopinath[a], Christian Desrosiers[b], Herve Lombaert[c]

[a,b,c] Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

Paper published in *GRAIL - Medical Image Computing and Computer Assisted Intervention*
(GRAIL - MICCAI), October 2020

Presentation

This chapter presents the article *"Graph Domain Adaptation for Alignment-Invariant Brain Surface Segmentation"* (Gopinath, Desrosiers & Lombaert, 2020a) submitted to **GRAIL-MICCAI** conference (Medical Image Computing and Computer Assisted Intervention), published on October 2020. The article was presented as an oral talk, held virtually at this conference and won honorable mention for the best paper award. The objective of this article is to develop a domain independent surface analysis model.

4.1 Introduction

The cerebral cortex is essential to a wide range of cognitive functions. Automated algorithms for brain surface analysis thus play an important role in understanding the structure and working of this complex organ. Nowadays, deep learning models such as convolutional neural networks (CNNs) provide state-of-the-art performance for most image analysis tasks, including image classification, registration, and segmentation (Arbabshirani *et al.*, 2017). However, these models typically require large annotated datasets for training, which are often expensive to obtain in medical applications. This limitation is especially true for the task of cortical segmentation, also known as *parcellation*, where generating ground truth data requires labeling possibly thousands of nodes on a highly-convoluted surface. This burden also explains why datasets for such tasks are relatively small. For instance, the largest publicly-available dataset for cortical parcellation, MindBoggle (Klein *et al.*, 2017), contains only 101 manually-annotated brain surfaces. Moreover, another common problem of deep learning models is their lack of robustness to differences in the distribution of training and test data. Hence, a CNN model trained on the data from a source domain usually fails to generalize to samples from other domains, i.e., the *target* domains.

Unsupervised domain adaptation (UDA) (Tajbakhsh, Jeyaseelan, Li, Chiang, Wu & Ding, 2020) has proven to be a powerful approach for making algorithms trained on source data generalize to examples from a target domain, without having explicit labels for these examples. Generative adversarial networks (GANs) (Goodfellow *et al.*, 2014) leverage adversarial training to produce realistic images. In this type of approach, a discriminator network classifies images produced by a generator network as real or fake, and the generator improves by learning to fool the discriminator. Following the success of GANs, adversarial techniques have later been proposed to improve the learning capability of CNNs across different domains. Adversarial domain adaptation methods for segmentation (Ghafoorian *et al.*, 2017; Javanmardi & Tasdizen, 2018; Vu *et al.*, 2019; Zhang *et al.*, 2017,1; Zou *et al.*, 2018) involve the concurrent training of two networks: a segmentator that learns to produce accurate segmentation outputs for labeled source examples, and a discriminator which forces the segmentator to have a similar prediction for examples of both source and target domains. These adversarial techniques usually rely on either feature space adaptation or output space adaptation. Initial works (Ganin & Lempitsky, 2015; Long *et al.*, 2015) focused on matching the distributions of features from source and target domain examples for classification tasks. As the output of CNNs for segmentation contains rich semantic information, (Tsai *et al.*, 2018) proposed a method that instead leverages output space adaptation. Various pixel-wise domain adaptation approaches have been developed for natural color images (Ganin & Lempitsky, 2015; Hoffman, Wang, Yu & Darrell, 2016). In medical image analysis, (Kamnitsas *et al.*, 2017a) proposed an adversarial neural network for MRI image

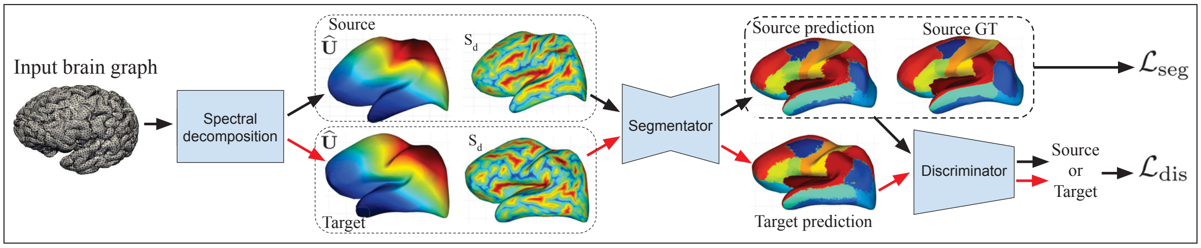


Figure 4.1 Overview of our architecture – The input brain graph is mapped to a spectral domain by decomposition of the graph Laplacian. The source and target domain are obtained by aligning the eigenbases to source reference and targets reference respectively. A segmentator GCN learns to predict a generic cortical parcel label for each domain. The discriminator aims at classifying the segmentator predictions, thereby assisting the segmentator GCN in adapting to both source and target domains.

segmentation which does not require additional labels on test examples from the target domain. Likewise, (Javanmardi & Tasdizen, 2018) presented a vessel segmentation approach for fundus images, which uses a gradient reversal layer for adversarial training. Recent work (Bateson *et al.*, 2019) also addressed the problem of domain adaptation by adding a differentiable penalty on the target domain. However, these domain adaptation techniques focus on data lying in the Euclidean space (natural or medical images) and, therefore, are not suitable for graph structures such as surface meshes.

The image space is inadequate to capture the varying geometry of the cerebral cortex. Differences in brain surface geometry hinder statistical frameworks from exploiting spatial information in Euclidean space. The extension of standard convolutions to non-Euclidean spaces like manifolds and graphs has led to the development of various geometric deep learning frameworks (Bronstein *et al.*, 2017; Monti *et al.*, 2017). A recent work (Cucurull *et al.*, 2018) proposed to use geometric deep learning for segmenting three cortical regions by relying on the spatial representation of the brain surface mesh. Later, based on the spectral representation of such meshes, (Gopinath *et al.*, 2019b) developed a graph convolution network (GCN) to parcellate the cerebral cortex. Despite offering more flexibility than Euclidean-based approaches, these methods are domain-dependent and would fail to generalize to new datasets (domains) without explicit re-training. Moreover, obtaining annotations for these new datasets is also challenging and time-consuming, due to the complexity of visualizing and labeling intricate surfaces.

In this paper, we address the limitations of existing techniques for cortical parcellation by proposing an adversarial domain adaptation method on surface graphs. Specifically, we focus on a problem shared by most GCN-based approaches, which is the need for a common basis to represent and operate on graphs. For approaches operating in Euclidean space, bringing surface graphs to this common basis usually involves transforming and possibly sub-sampling meshes to match a given reference, which is particularly difficult for convoluted surfaces like the cortex. As described in (Gopinath *et al.*, 2019b), this process can be greatly simplified by instead operating in the spectral domain, for instance using spectral GCNs (Bruna *et al.*, 2014b; Defferrard *et al.*, 2016). Nevertheless, spectral GNCs also need to perform some alignment to work. Hence, these models require computing the eigendecomposition of the graph Laplacian matrix to embed graphs in a space defined by a fixed eigenbasis. However, separate graphs may have different eigenbases, and the eigenvectors obtained for a given graph are only defined up to a sign and a rotation (if different eigenvectors share close eigenvalues). Due to these ambiguities, spectral GCNs cannot be used to compare multiple graphs directly and need an explicit alignment of graph eigenbases as an additional pre-processing step. Here, we focus on generalizing parcellation across multiple brain surface domains by removing the dependency on these domain-specific alignments.

The contributions of our work are multifold:

- We present, to the best of our knowledge, the first adversarial graph domain adaptation method for surface segmentation. Our novel method trains two networks in an adversarial manner, a fully-convolutional GCN segmentator and a GCN domain discriminator, both of which operate on the spectral components of surface graphs.

- Compared to existing approaches, our surface segmentation method offers greater robustness to differences in domain-specific alignment. Hence, our method yields a higher accuracy for non-aligned brain surfaces compared to a strategy without adversarial learning. Moreover, it also provides a better generalization for surfaces aligned to a different reference, without requiring an explicit re-alignment or manual annotations of these surfaces.
- We demonstrate the potential of our method for alignment-invariant parcellation of brain surfaces, using data from MindBoggle, the largest publicly-available manually-labeled surface dataset. Our results show significant mean Dice improvements compared to the same segmentation network without adversarial training and over a strong baseline approach based on Spectral Random Forest.

In the next section, we detail the fundamentals of our graph domain adaptation method for surface segmentation, followed by experiments validating the advantages of our method and a discussion of results.

4.2 Method

An overview of our proposed method is shown in Fig. 4.1. In the initial step, the cortical brain graph is embedded into the spectral domain using the graph Laplacian operator. Next, samples from the source domain only are aligned to a reference template using the Iterative Closest Point (ICP) algorithm. This algorithm works by repeating the following two steps until convergence: 1) mapping each node of the graph to align to its nearest reference node in the embedding space; 2) computing the orthogonal transformation (i.e., rotation and flip) which brings nodes nearest to their corresponding reference node. Since this process is iterative and external to the network architecture, it can be computationally expensive to run. However, we only need to apply it during training and, as shown in experiments, the proposed method can achieve good performance on non-aligned test examples by learning an alignment-invariant representation. Finally, a graph domain adaptation network is trained to perform alignment-independent parcellation. The segmentator network learns a generic mapping from input surface features, e.g. the spectral coordinates and sulcal depth of cortical points, to cortical parcel labels.

4.2.1 Spectral embedding of brain graphs

We start by describing the spectral graph convolution model used in this work. Denote as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ a brain surface graph with node set $\mathcal{V}$, such that $|\mathcal{V}| = N$, and edge set $\mathcal{E}$. Each node i has a feature vector $\mathbf{x}_i \in \mathbb{R}^3$ representing its 3D coordinates. We map $\mathcal{G}$ to a low-dimension manifold using the normalized graph Laplacian operator $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{A}$ is the weighted adjacency matrix and $\mathbf{D}$ the diagonal degree matrix. Here, we consider weighted edges and measure the weight between two adjacent nodes as the inverse of their Euclidean distance, i.e. $a_{ij} = (\|\mathbf{x}_i - \mathbf{x}_j\| + \epsilon)^{-1}$ where ϵ is a small positive constant. Letting $\mathbf{L} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$ be the eigendecomposition of $\mathbf{L}$, the normalized spectral coordinates of nodes are given by $\widehat{\mathbf{U}} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}$. The normalization with $\boldsymbol{\Lambda}^{-\frac{1}{2}}$ is used so that coordinates corresponding to smaller eigenvalues are given more importance in the embedding.

Denote the neighbors of node $i \in \mathcal{V}$ as $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$. The convolution operation used in our spectral GCN is defined as

$$\begin{aligned} z_{ip}^{(l)} &= \sum_{j \in \mathcal{N}_i} \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)}\, y_{jq}^{(l)}\, \varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j;\, \boldsymbol{\Theta}_k^{(l)}) \;+\; b_p^{(l)}, \\ y_{ip}^{(l+1)} &= \sigma\big(z_{ip}^{(l)}\big) \end{aligned} \tag{4.1}$$

where $y_{jq}^{(l)}$ is the feature of node j in the q-th feature map of layer l, $w_{pqk}^{(l)}$ is the weight in the k-th convolution filter between feature maps q and p of subsequent layers, $b_p^{(l)}$ is the bias of feature map p at layer l, and σ is a non-linear activation function. The information of the spectral embedding relating nodes i and j is included via a symmetric kernel $\varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j; \boldsymbol{\Theta}_k)$ parameterized by $\boldsymbol{\Theta}_k$. In this work, we follow (Gopinath *et al.*, 2019b) and use a Gaussian kernel: $\varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j; \boldsymbol{\mu}_k, \sigma_k) \;=\; \exp\big(-\sigma_k\, \|(\widehat{\mathbf{u}}_j - \widehat{\mathbf{u}}_i) - \boldsymbol{\mu}_k\|^2\big)$.

4.2.2 Graph domain adaptation

Our graph domain adaptation architecture contains two blocks: a segmentator GCN $\mathbf{S}$ performing cortical parcellation and a discriminator GCN D, which predicts if a given parcellation comes from a source or target graph. Let $\mathcal{X}_{\text{src}}$ be the set of source graphs and $\mathcal{X}_{\text{tgt}}$ the set of unlabeled domain graphs, with $\mathcal{X} = \mathcal{X}_{\text{src}} \cup \mathcal{X}_{\text{tgt}}$ the entire set of graphs available in training. In the first step, we optimize the segmentator GCN using labeled source graphs $\mathcal{G} \in \mathcal{X}_{\text{src}}$. We feed the segmentation network's prediction $\mathbf{S}(\mathcal{G})$ to the discriminator D whose role is to identify the input's domain (i.e., source or target). The gradients computed from an adversarial loss on target domain graphs are back-propagated from D to $\mathbf{S}$, forcing the segmentation to be similar for both the source and target domain graphs.

As in other adversarial approaches, we define the learning task as a minimax problem between the segmentator and discriminator networks,

$$\max_{D}\,\min_{S}\ \mathcal{L}(D,S) \ = \ \frac{1}{|\mathcal{X}_{\text{src}}|}\sum_{\mathcal{G}\in\mathcal{X}_{\text{src}}}\mathcal{L}_{\text{seg}}(S(\mathcal{G}),\mathbf{y}_{\mathcal{G}}) \ - \ \frac{\lambda}{|\mathcal{X}|}\sum_{\mathcal{G}\in\mathcal{X}}\mathcal{L}_{\text{dis}}\big(D(S(\mathcal{G})),z_{\mathcal{G}}\big), \tag{4.2}$$

where $\mathcal{L}_{\text{seg}}$ is the supervised segmentation loss on labeled source graphs, and $\mathcal{L}_{\text{dis}}$ is the discriminator loss on both source and target graphs, which is optimized in an adversarial manner for S and D.

Segmentator loss For each input graph, the segmentator network outputs a parcellation prediction $\widehat{\mathbf{y}}$ where $\widehat{y}_{ic}$ is the probability that node i belongs to parcel c. In this work, we define the supervised segmentation loss as a combination of weighted Dice loss and weighted cross-entropy (CE),

$$\mathcal{L}_{\text{seg}}(\widehat{\mathbf{y}},\mathbf{y}) \ = \ \left[1-\frac{\epsilon\,+\,2\sum_{i=1}^{N}\sum_{c=1}^{C}\omega_c\,y_{ic}\,\widehat{y}_{ic}}{\epsilon\,+\,\sum_{i=1}^{N}\sum_{c=1}^{C}\omega_c(y_{ic}+\widehat{y}_{ic})}\right] - \sum_{i=1}^{N}\sum_{c=1}^{C}\omega_c\,y_{ic}\,\widehat{y}_{ic}, \tag{4.3}$$

with y_{ic} being a one-hot encoding of the reference segmentation and ϵ a small constant to avoid zero-division. The weights ω_c balances the loss for parcels by increasing the importance given to smaller-sized regions. We follow (Gopinath *et al.*, 2019b) and set class weights ω_c as the total number of nodes divided by the number of nodes with label c. In the loss of Eq. (4.3), CE improves overall accuracy of node classification while Dice helps to have structured output for each parcel.

Discriminator loss Since the discriminator D is a domain classifier, we define its loss as the binary cross-entropy between its domain prediction (i.e., $\widehat{z} = 1$ for source or $\widehat{z} = 0$ for target):

$$\mathcal{L}_{\text{dis}}(\widehat{z}, z) \;=\; -(1 - z)\log(1 - \widehat{z}) \;-\; z\log\widehat{z}. \tag{4.4}$$

As mentioned before, this loss is maximized while updating the segmentator's parameters and minimized when updating the discriminator. Thus, the segmentator learns to produce surface parcellations that are alignment-invariant.

4.2.3 Network architecture

Segmentator: The segmentator is a fully-convolutional GCN comprised of 3 graph convolution layers with respective feature map sizes of 256, 128, and 32. At the input of the network, each node has 4 features: 3D spectral coordinates and an additional scalar measuring sulcal depth. All layers have $K_l = 6$ Gaussian kernels, similar to (Gopinath *et al.*, 2019b). Since the output has 32 parcels, our last layer size is set to 32. In the last layer, softmax operation is applied for parcellation prediction, and the remaining layers employ Leaky ReLU as an activation function to obtain filter responses in Eq. (4.1).

Discriminator: Similar to the segmentator network, we use 2 graph convolution layers, an average pooling layer, and 3 fully connected (linear) layers for classifying the segmentation domain. The first graph convolution layer takes segmentation predictions with 32 feature maps

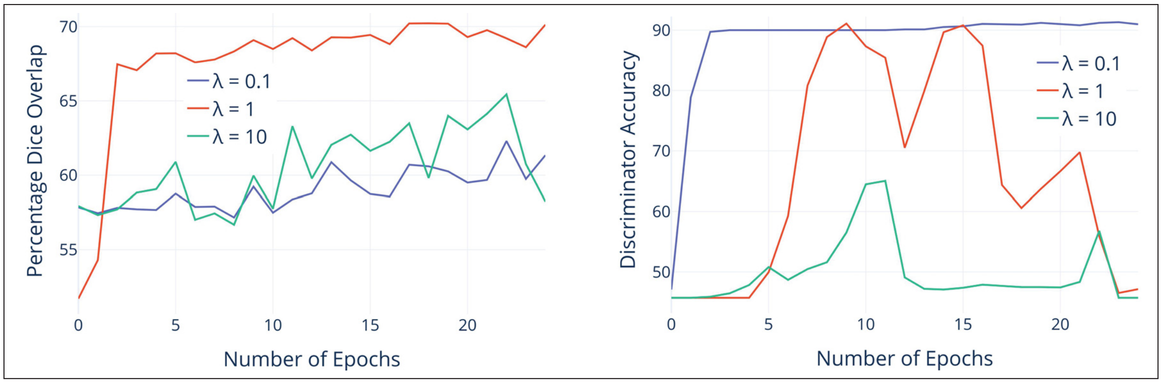


Figure 4.2 Effect of hyper-parameter λ – Segmentation performance in mean Dice (**left**) and Discriminator classification accuracy (**right**) on test examples, obtained for $\lambda \in \{0.1, 1, 10\}$.

as input. Moreover, the output sizes of the first two layers output are 128 and 64, respectively. Average pooling is used to reduce the input graph to a 1-D vector for the classification task. Three fully-connected layers are placed at the end of the network, with respective sizes of 32, 16, and 1. Each graph convolution layer has $K_l = 6$ Gaussian kernels. Sigmoid activation is applied to the last linear layer to predict the input domain of the graph sample and the remaining layers use Leaky ReLU.

4.3 Results

We evaluate the performance of our method using MindBoggle (Klein *et al.*, 2017), the largest manually-labeled brain surface dataset. This dataset contains the cortical mesh data of 101 subjects aggregated from multiple sites. Each brain surface includes 32 manually labeled parcels. We split this dataset into 70-10-20 training, validation and test sets. The training set has only 35 samples for the source and target domains each. To have more training samples and thus reduce overfitting, we sub-sample the node embeddings of each mesh to generate 25 examples of 10K nodes. This data augmentation technique, which is not possible in regular CNNs, is enabled by the spectral embedding of our approach.

Let P_c be the nodes predicted as having label $c \in \{1, \ldots, 32\}$, and G_c be the actual set of nodes with this label in the ground-truth parcellation. We evaluate performance using the mean Dice overlap:

$$\mathrm{MeanDice}(\mathbf{P}, \mathbf{G}) \;=\; \frac{1}{32}\sum_{c=1}^{32} \frac{2\,|P_c \cap G_c|}{|P_c| + |G_c|}. \tag{4.5}$$

All experiments were carried out on an i7 desktop computer with 16GB of RAM and an Nvidia Titan X 12 GB GPU. The code for our work is available at the following URL: https://tinyurl.com/yawdw7hh.

4.3.1 Effect of λ on parcellation

The loss function for adversarial training involves hyper-parameter λ, which controls the trade-off between parcellation accuracy on labeled source data and fooling the discriminator (i.e., alignment invariance). To assess the impact of this important hyper-parameter on performance, we show in Fig. 4.2 the segmentator mean Dice and discriminator classification accuracy on test examples at different training epochs, for $\lambda \in \{0.1, 1, 10\}$. As expected, when using a large $\lambda = 10$, the model focuses mostly on fooling the discriminator. This results in a low segmentation Dice, and a discriminator accuracy near 50% since the discriminator cannot distinguish between source and target parcellation outputs. Conversely, for a small $\lambda = 0.1$, the adversarial training gives less importance to fooling the discriminator, which translates in a high discriminator accuracy. However, this also leads to a poor performance on target examples, since the parcellation output for these examples differs greatly from those of source examples. This illustrates that a stronger adversarial learning is required to align the source and target domains. For the rest of our experiments, we selected $\lambda = 1$ based on the parcellation accuracy for *validation* examples.

Table 4.1 Comparison with surface segmentation approaches – Mean Dice and standard deviation on test data. The first result column corresponds to the default setting where test (i.e., target domain) graphs are not aligned. For the second column, test graphs were aligned on the same reference as training (i.e., source domain) graphs. Result columns 3-7 correspond to the setting where all test graphs are aligned to four randomly-selected target graphs (a different graph for each column). Bold font highlights a performance statistically higher than all other methods (t-test p < 0.01).

		Alignment to reference graph				
Method	**No alignment**	**Source**	**Rand. target 1**	**Rand. target 2**	**Rand. target 3**	**Rand. target 4**
Spectral RF (Lombaert *et al.*, 2015b)	65.4 ± 9.0	81.9 ± 3.4	60.0 ± 1.8	55.3 ± 2.1	60.2 ± 4.0	55.2 ± 3.0
Seg-GCN (Gopinath *et al.*, 2019b)	71.4 ± 7.9	86.5 ± 2.8	67.8 ± 2.0	58.8 ± 2.8	63.5 ± 3.2	60.1 ± 3.6
Adv-GCN (ours)	**73.8 ± 6.0**	85.7 ± 3.5	**73.5 ± 2.0**	**72.5 ± 2.6**	**72.4 ± 2.4**	**71.7 ± 3.3**

4.3.2 Comparison with the state-of-the-art

We next compare our method, called Adv-GCN in the following results, against two other graph-based approaches for surface parcellation. This first one is the Spectral Random Forest (RF) algorithm proposed in (Lombaert *et al.*, 2015b), which performs the same spectral graph embedding as our method, and then uses the spectral coordinates and sulcal depth at individual nodes to train a RF classifier. As done in (Lombaert *et al.*, 2015b), we employed 50 trees to build the RF model. This comparison baseline was included to show the limitation of point-based approaches which ignore the relationship between nodes when predicting labels. The second approach, called Seg-GCN, is the same segmentation GCN as in our method, but trained without the adversarial loss. For this baseline, which is similar to the method presented in (Gopinath *et al.*, 2019b), our goal is to show the benefit of learning an alignment-invariant representation with adversarial domain adaptation.

The surface parcellation approaches are compared in three different test settings. In the first one, the approaches are applied on target examples without any alignment. This corresponds to the normal application setting of our alignment-invariant method. For the second one, we align all target examples on the same reference surface as the one used for source examples.

This setting requires to retain the reference surface and apply ICP alignment in pre-processing for each test surface. Finally, in the third setting, target examples are aligned to a reference surface chosen randomly in the test set. This last setting corresponds to the case where we want to parcel surfaces from a dataset which was processed differently than the source dataset.

Results of this experiment are summarized in Table 4.1. When test examples are aligned to the same source reference (i.e., no domain shift), our segmentation GCN architecture, with or without adversarial learning, outperforms Spectral RF by a large margin. This illustrates the importance of considering the relationship between different nodes in the graph, as in our graph convolution model. However, when applied to non-aligned test surface, our Adv-GCN method achieves a 2.4% improvement in mean Dice over Seg-GCN, and 8.4% over Spectral RF. This demonstrates the benefit of learning an alignment-invariant representation via adversarial domain adaptation. Furthermore, the improvement provided by our Adv-GCN method is even more significant for surfaces aligned to a random target reference (last four columns of Table 4.1). Thus, across the four random target references, Adv-GCN yields an average improvement of 14.9% compared to Spectral RF and 10.0% compared to Seg-GCN. This shows the strength of adversarial learning to match the output distribution for two fixed domains.

The average Dice overlap for individual parcels is shown in Fig. 4.3. As can be seen, Adv-GCN provides a higher mean and smaller variance for most of the 32 parcels. By inspecting results, we find that accuracy is correlated with parcel size, with larger parcels generally better segmented than smaller ones. Figure 4.4 shows qualitative results for different graph segmentation methods. As highlighted by the red circle, our Adv-CGN gives a more accurate segmentation compared to Seg-GCN and Spectral RF, with an improvement over 13% in parcel-averaged Dice.

4.4 Discussion and Conclusion

In this paper, we presented a novel adversarial domain adaptation framework for brain surface parcellation. The proposed algorithm leverages an adversarial training mechanism to obtain

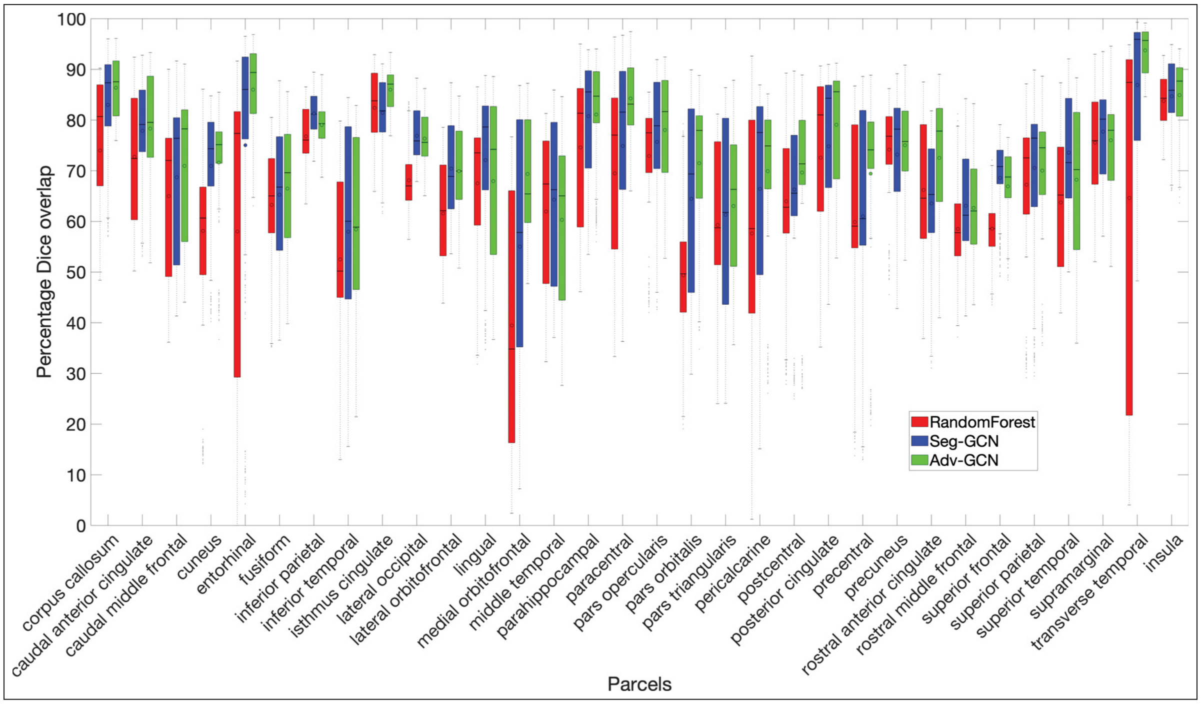


Figure 4.3 Segmentation Dice for individual parcels – Box-plot of mean Dice overlap achieved by three different methods for all 32 cortical parcels when *test subjects are not aligned*.

an alignment-invariant surface segmentation, and overcomes the limitations of spectral GCNs (Bruna *et al.*, 2014b; Defferrard *et al.*, 2016) that require finding an explicit alignment of graph eigenbases. Table 4.1 shows a clear improvement in performance over the same spectral GCN without adversarial training (Seg-GCN) and the Spectral Random Forest (RF) algorithm (Lombaert *et al.*, 2015b). Specifically, our method yields a 2.4% mean Dice improvement over Seg-GCN and 8.4% over Spectral RF, for non-aligned test surfaces. This improvement reaches over 10% for test surface aligned to a random target reference. Qualitative results in Fig. 4.4 illustrate the better parcellation of our method for non-aligned surfaces.

In some experiments, we observed a tendency of the discriminator to overfit the training set, which impeded domain adaptation in the learning process. In a future study, two strategies could be explored to overcome this problem: using other types of discriminator, for instance the Least Squares GAN (Mao, Li, Xie, Lau, Wang & Paul Smolley, 2017) or Wasserstein GAN

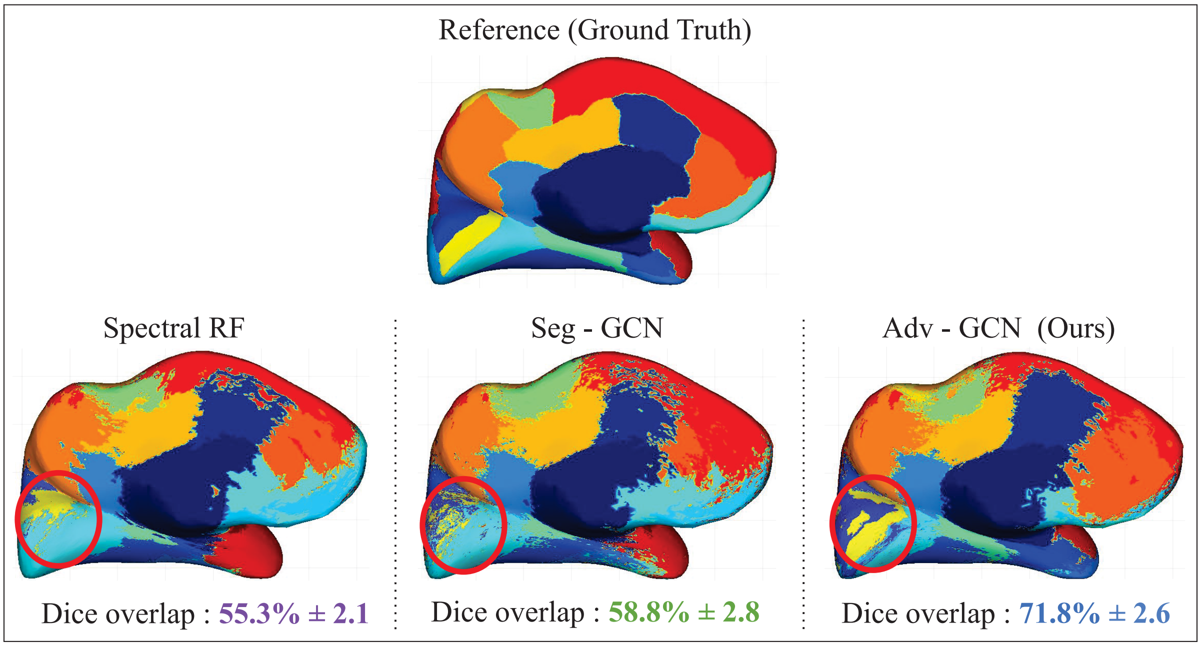


Figure 4.4 Qualitative comparison of parcellation – Parcellation outputs of the three surface segmentation approaches for a single non-aligned test surface. For better visualization, segmented parcels are drawn on an inflated surface. For each approach, we report the average Dice and standard deviation computed over the 32 parcels. As highlighted by the red circle, our adversarial GCN (Adv-CGN) gives a more accurate segmentation compared to the same model without adversarial training (Seg-GCN) and Spectral Random Forest (RF).

(Arjovsky, Chintala & Bottou, 2017), and applying data augmentation on labeled brain surface meshes. While our adversarial graph domain adaptation technique was demonstrated on cortical parcellation, it also has potential for other surface segmentation problems where a domain shift is present. Likewise, our method could be useful for semi-supervised surface segmentation, thereby mitigating the need for large amounts of labeled surfaces. In this setting, the same architecture could be used, however the discriminator would predict if the segmentation output is for a labeled or unlabeled example from the same domain. We plan to evaluate the impact of higher frequency input representations with performance measures such as Hausdorff distance in future work.

CHAPTER 5

LEARNING JOINT SURFACE RECONSTRUCTION AND SEGMENTATION, FROM BRAIN IMAGES TO CORTICAL SURFACE PARCELLATION

Karthik Gopinath[a], Christian Desrosiers[b], Herve Lombaert[c]

[a,b,c] Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

Paper submitted for review in *Journal of Medical Image Analysis* (MedIA), August 2021

Presentation

This chapter presents the article *"Learning Joint Surface Reconstruction and Segmentation, from Brain Images to Cortical Surface Parcellation"* submitted to the**MedIA** journal (Medical Image Analysis). The article was initially accepted for publication at **MICCAI** (Medical Image Computing and Computer Assisted Intervention) 2021, held virtually. The objective of this article is to jointly reconstruct and segment cortical surface from input MRI volume.

5.1 Introduction

Brain surface analysis requires the accurate reconstruction and segmentation of cortical surfaces from MRI volumes (Glasser *et al.*, 2016; Querbes, Aubry, Pariente, Lotterie, Démonet, Duret, Puel, Berry, Fort, Celsis et al., 2009). Standard surface processing pipelines for reconstructing cortical surfaces (Dahnke, Yotter & Gaser, 2013; Fischl *et al.*, 2004b; Kim, Singh, Lee, Lerch, Ad-Dab'bagh, MacDonald, Lee, Kim & Evans, 2005; Kriegeskorte & Goebel, 2001; Shattuck & Leahy, 2002; Styner *et al.*, 2006) 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 cortical surfaces are segmented into neuroanatomical parcels in a subsequent and highly-expensive step. Such segmentation can take several hours to finish, which involves the re-projection of each surface to a sphere via a metric-preserving inflation process, registration to a spherical atlas (Fischl *et al.*, 1999; Klein & Tourville, 2012) and cortical parcellation using atlas labels (Desikan, Ségonne, Fischl, Quinn, Dickerson, Blacker, Buckner, Dale, Maguire, Hyman et al., 2006).

Recently, Henschel *et al.* (2020) developed a framework called FastSurfer using deep learning that accelerates the processing times for brain segmentation and spectral embedding for registration to a spherical atlas. Despite reducing computation times considerably, this pipeline still performs reconstruction and segmentation in two consecutive steps. To overcome this limitation, Cruz *et al.* (2021) proposed a deep learning model called DeepCSR for cortical surface reconstruction. This method draws inspiration from (Park *et al.*, 2019), and samples points on a reference grid of arbitrary resolution to reconstruct a surface without the need for an explicit segmentation step. However, this process is highly expensive in terms of both computation and memory for detailed surfaces with hundreds of thousands of points. Additionally, DeepCSR only performs surface reconstruction, and cannot be used for parcellation which is one of the most time-costly operations in conventional neuroimaging pipelines (Fischl *et al.*, 2004b). Approaches that directly operate (López-López, Vázquez, Poupon, Mangin, Ladra & Guevara, 2020; Wu, Zhao, Xia, Wang, Lin, Gilmore, Li & Shen, 2019) or learn on surface data (Gopinath *et al.*, 2019b,2; Lombaert *et al.*, 2015b) have been used for cortical parcellation, but are designed to process single surfaces separately for each subjects. Spectral embeddings of surface meshes in a low-dimensional space can be exploited to predict cortical parcellation labels Germanaud, Lefèvre, Toro, Fischer, Dubois, Hertz-Pannier & Mangin (2012); Lombaert *et al.* (2015b). However, a major limitation of these early learning approaches is that mesh nodes are considered separately instead of jointly. Recent work has proposed using graph convolutional networks (GCN) (Gopinath *et al.*, 2019b,2; He, Gopinath, Desrosiers & Lombaert, 2020; Wu *et al.*, 2019) to exploit the connectivity information of a mesh graph. While such strategy provides an accurate and faster parcellation of the cortical surface, it is highly sensitive to the quality of the surface reconstruction step. Hence, small errors or holes in the reconstructed cortical mesh may cause to parcellation to fail.

We propose SegRecon, a novel deep learning model 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. A CNN based on 3D-UNet (Çiçek *et al.*, 2016) is used to predict, for each voxel of an input volume, the brain hemisphere containing the voxel, its signed distance to the nested surfaces (white matter and pial surfaces) of that hemisphere, used for surface reconstruction, and the spherical coordinates in the registered atlas space, used for surface parcellation. By learning to solve this multi-task problem, the network can be used to reconstruct and segment brain surfaces efficiently and in a topologically-accurate manner (Bazin & Pham, 2007).

The main contributions of our work are the following:

- To our knowledge, we propose the first deep learning model for the joint reconstruction of multiple nested surfaces and their segmentation, with an application on brain surfaces. This contrasts with existing approaches, which either perform surface reconstruction and segmentation in separate steps (Henschel *et al.*, 2020), are limited to reconstruction (Cruz *et al.*, 2021), or require a pre-generated mesh as input (Gopinath *et al.*, 2019b; Lombaert *et al.*, 2015b; Wu *et al.*, 2019);

- Compared to the current surface reconstruction learning approaches (such as DeepCSR), the proposed method implements a fully-convolutional architecture that densely predicts the location of all input voxels relative to cortical surfaces, in a single feed-forward pass. Our method also leverages a novel surface reconstruction loss that controls the minimum and maximum distance between white matter and pial surfaces (i.e., cortical thickness), thereby ensuring that these surfaces never cross;

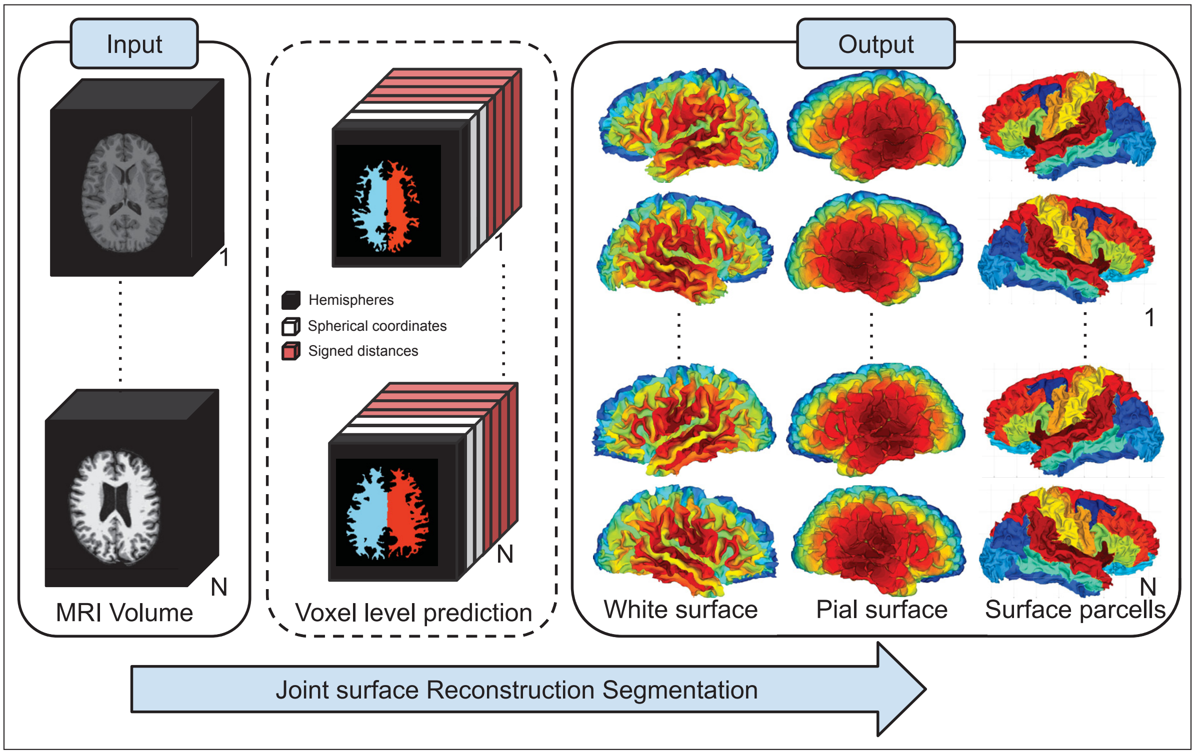


Figure 5.1 Overview of surface reconstruction segmentation algorithm – Our proposed method for joint brain surface reconstruction and segmentation from images. On the left is input: MRI volume X. In the middle are the intermediate learned voxel level predictions from the 3D CNN model. These include for each voxel v: hemisphere label $h_v(X)$, azimuthal angle $\gamma_v(X)$ and polar angle $\phi_v(X)$ in the spherical atlas space, signed distance to white matter surface $d_v^w(X)$, and signed distance to pial surface $d_v^p(X)$. On the right are the reconstructed white and pial surface along with cortical parcels for each hemisphere of the brain.

- We present a comprehensive set of experiments involving three publicly-available datasets, i.e., MindBoggle (Klein *et al.*, 2017), OASIS (Marcus, Wang, Parker, Csernansky, Morris & Buckner, 2007) and ABIDE-I (Di Martino, Yan, Li, Denio, Castellanos, Alaerts, Anderson, Assaf, Bookheimer, Dapretto et al., 2014), that compare the surface reconstruction and segmentation accuracy of our method against several baselines. Our results demonstrate the major advantages of our method over standard brain surface analysis pipelines are recent approaches for cortical parcellation. With respect to the widely-used FreeSurfer software, our method generates surfaces with an average Hausdorff distance less than 0.52 mm, while boosting the parcellation Dice by 4.3% and being several orders of magnitude faster.

In the next section, we present our proposed joint reconstruction and segmentation approach, describing in detail the network architecture, training losses and inference steps. The performance of our method is then evaluated on the MindBoggle (Klein *et al.*, 2017), OASIS (Marcus *et al.*, 2007) and ABIDE-I (Di Martino *et al.*, 2014) datasets. The ablation study and comparison to the state-of-art in our experiments demonstrate the important benefits of our method.

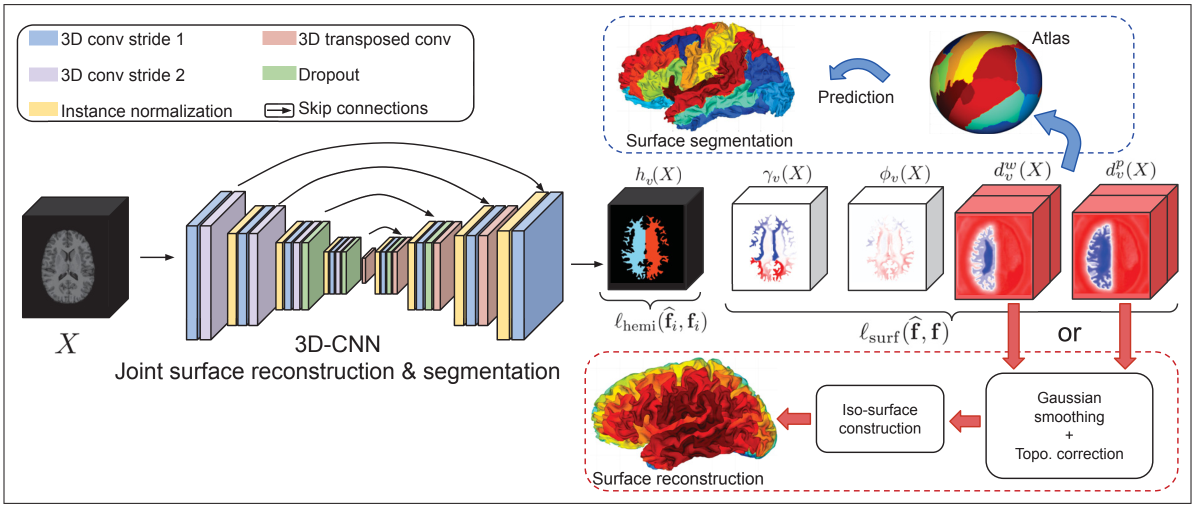


Figure 5.2 Overview of SegRecon architecture – The 3D-CNN model takes as input MRI volume X for joint learning of the signed distance to white-to-grey matter interface, grey-matter-to-CSF interface and its corresponding spherical coordinates in the registered atlas space. (*Red*) The cortical white matter or pial surface is reconstructed by applying Gaussian smoothing and topological correction on the predicted signed distance map prediction $d_v^w(X)$ or $d_v^p(X)$, 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 of the left hemisphere white matter surface is shown here.

5.2 Method

An overview of SegRecon is shown in Fig. 5.2 with the end-to-end surface construction and segmentation steps illustrated. Let $\mathcal{D} = \{(X_i, S_i^W, S_i^P, Y_i)\}_{i=1}^n$ be a training set where each example is composed of: a 3D volume $X_i \in^{|\Omega|}$ with voxel set $\Omega \subset \mathbb{Z}^3$, a white matter surface $S_i^W \in^{m\times 3}$ defined by m points, a pial surface $S_i^P \in^{n\times 3}$ defined by n points, and a one-hot encoded segmentation $Y_i \in [0,1]^{m\times c}$ of the white matter surface, where c is the number of segmentation classes. The goal is to learn a model f parameterized by θ which maps an input 3D volume X to a white matter surface S^W with corresponding parcel labels Y, and a pial surface S^P.

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^W and S^P, where the number of points can vary from one surface to another and points can lie anywhere in 3D space. In (Cruz *et al.*, 2021), this problem is solved by giving as input to model f both the image X and a query point $p \in^3$ 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

$$f_v(X) \;=\; \left[d_v^w, d_v^p, \phi_v, \gamma_v, h_v^{lh}, h_v^{rh}, h_v^{bg}\right], \tag{5.1}$$

where d_v^w is the signed distance from v to its nearest surface point, such that $d_v^w \leq 0$ if v is inside the surface else $d_v^w > 0$. Similarly, d_v^p is the signed distance from v to its nearest pial surface. ϕ_v, γ_v are the polar angle and azimuthal angle of $v \in \Omega$ defining its position in the spherical atlas, and h_v^{lh}, h_v^{rh}, $h_v^{bg} \in [0,1]$ 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 (Bazin & Pham, 2007) over the predicted surface points prevents the extraction of critical points yielding topological defects. The resulting white and pial surfaces are defined implicitly as the 0-levelset of their respective distance map and can be efficiently reconstructed using an iso-surface extraction algorithm such as the Marching Cubes (Lorensen & Cline, 1987).

5.2.1 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

$$\mathcal{L}(\theta; \mathcal{D}) \;=\; \sum_{i=1}^{n} \ell_{\mathrm{wsurf}}(\widehat{\mathbf{f}}_i, \mathbf{f}_i) \;+\; \ell_{\mathrm{psurf}}(\widehat{\mathbf{f}}_i, \mathbf{f}_i) \;+\; \lambda_1\, \ell_{\mathrm{hemi}}(\widehat{\mathbf{f}}_i, \mathbf{f}_i) \;+\; \lambda_2\, \ell_{\mathrm{thick}}(\widehat{\mathbf{f}}_i), \tag{5.2}$$

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

$$\begin{aligned}\ell_{\mathrm{wsurf}}(\widehat{\mathbf{f}}, \mathbf{f}) \;=\; \sum_{v \in \Omega} \mathbb{1}_{|d_v^w| \leq \epsilon} \cdot \Big[(\widehat{d}_v^w - d_v^w)^2 \;+\; \min\big\{(\widehat{\phi}_v - \phi_v)^2,\, (1 + \widehat{\phi}_v - \phi_v)^2\big\} \\ +\; \min\big\{(\widehat{\gamma}_v - \gamma_v)^2,\, (1 + \widehat{\gamma}_v - \gamma_v)^2\big\}\Big].\end{aligned} \tag{5.3}$$

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 white matter surface point in order to focus learning on relevant points close to our surface. This is achieved with function $\mathbb{1}_{|d_v^w| \leq \epsilon}$ in Eq. (5.3). 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^w is, therefore, defined between the center of the voxel v in image space and the nearest point on white matter surface S^W. In this work, we use the white matter surface mesh generated by FreeSurfer for training. The sign of d_v^w 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 (Fischl *et al.*, 2004b) with the Desikan-Killiany-Tourville (DKT) atlas (Klein & Tourville, 2012).

The second loss term, ℓ_{psurf}, ensures that the signed distance of voxels to the nearest pial surface is predicted accurately. We define it as

$$\ell_{\text{psurf}}(\widehat{\mathbf{f}}, \mathbf{f}) \;=\; \sum_{v \in \Omega} \mathbb{1}_{|d_v^p| \leq \epsilon} \cdot (\widehat{d_v^p} - d_v^p)^2. \tag{5.4}$$

where d_v^p is the distance defined between the center of the voxel v and its closest point on pial surface S^P obtained by FreeSurfer pial meshes. The sign of the distance d_v^p is estimated using the brain segmentation mask with voxels inside the brain mask having negative distance. Similar to ℓ_{wsurf} in Eq. (5.3), $\mathbb{1}_P$ is the indicator function used to restrict the training to the useful voxels within a distance of ϵ to the closest pial surface.

The third term, ℓ_{hemi} enables the network to predict if a voxel v lies in the left hemisphere (lh), in the right hemisphere (rh) or is outside both (bg). This prediction is necessary since the surface atlas is defined separately for each hemisphere. Here, we use cross-entropy as loss function:

$$\ell_{\text{hemi}}(\widehat{\mathbf{f}}, \mathbf{f}) \;=\; -\sum_{v \in \Omega} \sum_{c \in \{lh, rh, bg\}} h_v^c \, \log \widehat{h}_v^c. \tag{5.5}$$

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

Since the white matter and pial surfaces are reconstructed from two separate predictions, it may happen that predicted surfaces are near to the ground-truth while still violating anatomical constraints. For example, in very thin regions of the cortex, the reconstructed surfaces may overlap or even cross each other. To avoid this problem, we add a last term to the loss function,

ℓ_{thick}, which controls the minimum and maximum distance between the surfaces:

$$\ell_{\text{thick}}(\widehat{\mathbf{f}}) \;=\; \sum_{v\in\Omega} \mathbb{1}_{d_v^p \le 0} \cdot \big[\, \max(\widehat{d}_v^p - \widehat{d}_v^w + t_{min}\,, 0) \;+\; \max(\widehat{d}_v^w - \widehat{d}_v^p - t_{max}\,, 0)\big] \tag{5.6}$$

where t_{min} and t_{max} are the minimum and maximum allowed inter-surface distances (cortical thickness). These hyperparameters can be set based on the dataset ground-truth or some clinical reference. For instance, a cortical thickness range from 1 to 4.5 mm is reported in (Fischl & Dale, 2000). We use similar values in this work: $t_{min} = 0$ and $t_{max} = 5$. Effectively, this prevents surfaces from crossing each other or separating beyond 5 mm. As defined in Eq. (5.6), this penalty is only calculated for voxels inside the pial surface, i.e., voxels v such that $\mathbb{1}_{d_v^p \le 0}$.

5.2.2 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 white matter surface distance map $\widehat{d}^w$ using a single convolution operation and employ a topological correction step (Bazin & Pham, 2007) to overcome any defects in the surface. The same steps are followed to extract the 3D pial surface using distance map $\widehat{d}^p$.

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

$$S^c \;=\; \big\{ v \in \Omega \mid |d_v| \le \epsilon \,\wedge\, c = \arg\max_{c'} \,\widehat{h}_v^{c'} \big\}, \;\; c \in \{lh, rh\}. \tag{5.7}$$

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.

5.2.3 Implementation details

The overall architecture of SegRecon is shown in Fig. 5.2. As an input, we provide the skull-stripped, intensity normalized 3D T1-MRI volume. We use a 3D-UNet architecture similar to (Çiçek *et al.*, 2016) 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 two output channels produce the signed distance map $\widehat{d}_v^w$ and $\widehat{d}_v^p$ for each voxel v. The network parameters, θ, are optimized using a stochastic gradient descent with the Adam optimizer (Kingma & Ba, 2014). During training, we pick the maximum distance of surface voxels in Eq. (5.3) to be $\epsilon = 2.5$, which corresponds to the overall average thickness reported in (Fischl & Dale, 2000). The surface is reconstructed using the Marching Cubes algorithm (Lorensen & Cline, 1987) on the 0-levelset of its predicted signed distance map, smoothed with a Gaussian kernel of sigma$=$0.5 and topologically corrected with the method of (Bazin & Pham, 2007). We use an i7 desktop machine with 16Gb RAM and Nvidia RTX 2080 GPU for our work.

5.3 Experiments and results

To benchmark the performance of our method, we use one of the largest publicly-available dataset containing manual surface parcellation, MindBoggle (Klein *et al.*, 2017). 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%. We also use the ABIDE-I (Di Martino *et al.*, 2014) and OASIS (Marcus *et al.*, 2007) databases as independent test sets to measure the surface reconstruction error of our method with FreeSurfer-generated cortical and pial surfaces. The ABIDE dataset contains brain surfaces for 1035 subjects with 530 healthy and 505 autism spectrum disorder (ASD) subjects. Likewise, the OASIS dataset comprises a total of 226 brain surfaces from 93 healthy subjects and 133 subjects with Alzheimer's disease (AD). These two datasets are used to validate the robustness of the method to various factors, including image acquisition and processing parameters, age (the majority of ABIDE subjects are children or adolescents, while most OASIS subjects are elders) and cortical surface alterations (AD subjects in OASIS have a thinner cortex, on average, resulting from the neurodegenerative disease).

In a first experiment, we validate the benefit of predicting a signed distance map for surface reconstruction compared to predicting a binary mask. For this experiment, we train the model using only data from MindBoggle, and measure the reconstruction error on subjects from the ABIDE and OASIS datasets. The qualitative results of the reconstructed surface are shown in Fig. 5.3 and 5.4. In the next experiment, we evaluate the impact 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 our joint reconstruction and parcellation model against state-of-the-art methods.

Table 5.1 Performance of surface reconstruction for white surface – The reconstruction error (*mm*) measured between white surface meshes generated by our SegRecon method and FreeSurfer generated meshes. The chamfer distance (CD), absolute average distance (AAD) and Hausdorff distance (HD) are reported for white surface meshes of both left and right hemisphere. The first row highlights the performance of our method with surface reconstruction from signed distance (SD) map, where as, the second column shows the limitation of using binary (BW) segmentation map for surface reconstruction. The third and fourth rows report the reconstruction error on OASIS dataset. Likewise, the last two rows shows the results on ABIDE dataset. Reconstruction metrics are reported on unseen OASIS and ABIDE dataset containing both healthy and AD/ASD subjects.

Experiment	Left White Matter			Right White Matter		
	CD	AAD	HD	CD	AAD	HD
SD map	1.909 ± 0.25	0.449 ± 0.04	0.895 ± 0.08	1.944 ± 0.26	0.465 ± 0.05	0.936 ± 0.11
BW map	5.019 ± 0.35	0.630 ± 0.04	3.104 ± 0.60	4.954 ± 0.27	0.632 ± 0.04	3.032 ± 0.47
OASIS CN	2.229 ± 1.13	0.515 ± 0.06	0.967 ± 0.16	2.083 ± 0.74	0.503 ± 0.06	0.957 ± 0.14
OASIS AD	2.074 ± 0.81	0.488 ± 0.09	0.921 ± 0.20	2.039 ± 0.76	0.483 ± 0.07	0.913 ± 0.15
ABIDE CN	2.020 ± 0.77	0.437 ± 0.06	0.900 ± 0.15	2.011 ± 0.76	0.442 ± 0.05	0.912 ± 0.15
ABIDE ASD	2.081 ± 0.92	0.448 ± 0.09	0.927 ± 0.29	2.055 ± 0.88	0.452 ± 0.08	0.937 ± 0.31

Table 5.2 Performance of surface reconstruction for pial surface – The reconstruction error (*mm*) measured between pial surface meshes generated by our SEGRECON method and FreeSurfer generated meshes. Reconstruction metrics are reported on unseen OASIS and ABIDE dataset containing both healthy and AD/ASD subjects. The chamfer distance (CD), absolute average distance (AAD) and Hausdorff distance (HD) are reported for pial surface meshes of both left and right hemisphere.

Experiment	Left Pial			Right Pial		
	CD	AAD	HD	CD	AAD	HD
SD map	1.908 ± 0.25	0.435 ± 0.04	0.875 ± 0.08	1.944 ± 0.26	0.453 ± 0.03	0.942 ± 0.09
BW map	5.034 ± 0.34	0.806 ± 0.04	2.730 ± 0.60	4.970 ± 0.27	0.829 ± 0.04	2.509 ± 0.47
OASIS CN	2.226 ± 1.13	0.422 ± 0.07	1.037 ± 0.30	2.080 ± 0.74	0.429 ± 0.05	1.032 ± 0.16
OASIS AD	2.076 ± 0.82	0.429 ± 0.06	1.042 ± 0.21	2.037 ± 0.75	0.443 ± 0.05	1.058 ± 0.14
ABIDE CN	2.022 ± 0.78	0.568 ± 0.09	1.183 ± 0.30	2.011 ± 0.75	0.574 ± 0.08	1.215 ± 0.30
ABIDE ASD	2.079 ± 0.91	0.572 ± 0.10	1.216 ± 0.42	2.055 ± 0.88	0.580 ± 0.11	1.263 ± 0.46

5.3.1 Surface Reconstruction

To assess the quality of reconstructed surfaces, we use the Chamfer distance (CD) (Park *et al.*, 2019), absolute average distance (AAD) (Cruz *et al.*, 2021) and Hausdorff distance (HD) (Cruz *et al.*, 2021). Chamfer distance is a widely-used evaluation metric defined as the sum between the average squared-distance from predicted surface points to their nearest point on the ground truth surface and the average squared-distance from ground truth surface points to their nearest point on the predicted surface. Similarly, AAD measures the mean *absolute* nearest-neighbor distance between the predicted and ground-truth surface points, averaging values computed in both directions (predicted to ground-truth and the opposite). Finally, HD computes the maximum distance between a point on a surface and its nearest point on the other surface. As in (Cruz *et al.*, 2021), to minimize the impact of outliers, we consider the 90^{th} percentile of nearest-neighbor distances, and keep the maximum between values computed for one direction and the other. For all metrics, a lower value in *mm* indicates a better surface reconstruction.

We first evaluate the benefit of using a signed distance (SD) map, when reconstructing the white and pial cortical surfaces, by comparing it against using a binary mask (BW). To predict the binary mask, we use an architecture similar to the one in Fig. 5.2 where the last two output maps (corresponding to white matter and pial surfaces) are generated with sigmoid activations. As reported in Table 5.1, an improvement in CD from 5.0 mm to 1.9 mm is obtained when signed distance map is used for white matter surface reconstruction. A similar improvement over the binary mask approach is also observed in terms of AAD and HD. Qualitative results, presented in Fig. 5.3, show that the meshes reconstructed using signed distance maps are more regular and closer to FreeSurfer-generated meshes, compared to those obtained with binary masks.

Our surface reconstruction method was also tested on the OASIS and ABIDE datasets, not used for training, to evaluate its robustness. As can be found in Table 5.1, our method obtained a mean AAD below 0.52 mm and mean HD less than 0.97 mm for the white matter surface, in both datasets. Similarly, reconstructed pial surfaces in Table 5.2, on both datasets have a mean AAD no greater than 0.58 mm and mean HD less than 1.27 mm for pial surfaces in both datasets. These results, obtained for subjects of very different ages and with cortical alterations, are comparable to those obtained for the MindBoggle test set. The qualitative results in Fig. 5.4 validate the visual similarity in surface reconstruction of our method, across datasets.

5.3.2 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 5.3 reports the mean Dice

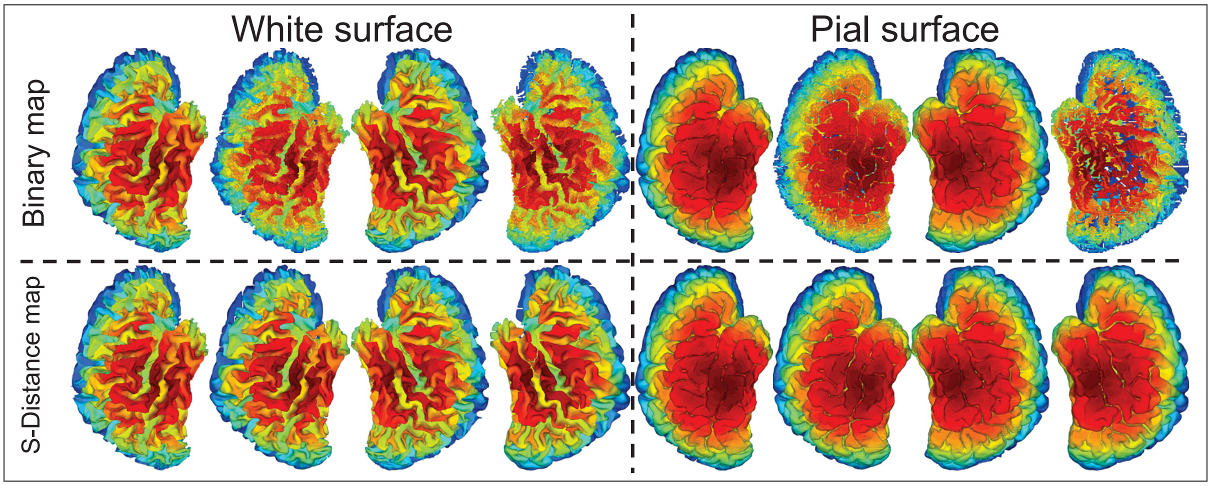


Figure 5.3 Surface reconstruction visualization – Comparison of cortical white and pial surface reconstructed by our method (even columns) with FreeSurfer generated surfaces (odd columns). The first row shows the surfaces reconstructed by our method using binary mask prediction. A reconstruction error of 2.8 mm in Hausdorff distance (HD) is seen with the use of binary mask (BW) vs. a reconstruction error of 0.9 mm in Hausdorff distance (HD) with the use of a signed distance (SD) map. The reconstruction in the last column for pial surface highlights the downside of using BW masks with irregular surface reconstruction.

Table 5.3 Effect of reference atlas on parcellation – 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**

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 5.3 (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.

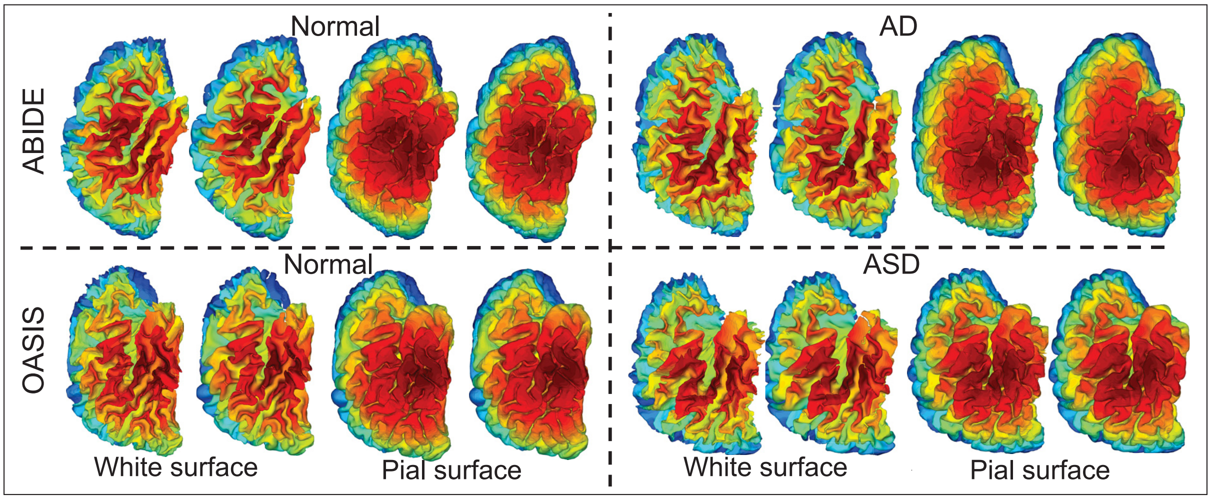


Figure 5.4 Reconstruction on unseen ABIDE and OASIS datasets – Comparison of a cortical surface predicted by our joint reconstruction and segmentation method and FreeSurfer (Fischl *et al.*, 1999). Our SegRecon method yields visually similar results while being orders of magnitude faster. Reconstruction error on the unseen ABIDE and OASIS dataset with both healthy and ASD/AD subjects are identical to the MindBoggle dataset the model is trained on, indicating the robustness of the proposed method. Only right hemisphere is shown here.

5.3.3 Comparison with the state-of-the-art

We next compare our joint reconstruction and parcellation method SegRecon against several baselines and recent approaches for these tasks. Table 5.4 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 5.4, gives a low Dice score of 79.95%. As mentioned above, this is due to the greater number of network outputs (i.e., one output per class) compared to simply predicting the two spherical atlas coordinates.

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) (Lombaert *et al.*, 2015b), 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) (Gopinath *et al.*, 2019b). 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 only 8 seconds per volume. That is a 4.30% improvement over the Dice score of FreeSurfer, at a fraction of its computational cost.

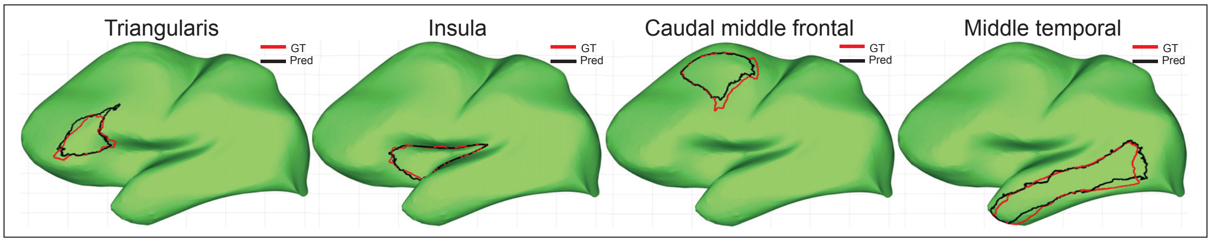


Figure 5.5 Visualization of parcellation performance – The manual parcellation boundaries are shown in red, with our predicted parcellation boundaries in black. Our model segments 32 parcels in total on the brain surface. We show four parcels, namely, Triangularis, Insula, Caudal middle frontal and middle temporal of the left hemisphere for qualitative analysis. The cortical mesh is inflated here for visualization.

Next, we evaluate the performance of our SegRecon method in two different settings. First, we show the importance of the hemisphere prediction loss of Eq. (5.5) on performance. To do so, we reduce the weight of the loss term ℓ_{hemi} in Eq. (5.2) to $\lambda = 0.0001$ during training. This ablation baseline is denoted as *w/o hemisphere* in Table 5.4. As can be observed, the lack of accurate hemisphere prediction results in ambiguous label prediction for surface voxels in both hemispheres which leads to 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 ($\widehat{\phi}_v$ 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. 5.5, where we illustrate the differences between the predicted and manual label boundaries for four different parcels or regions.

Table 5.4 Evaluation of SegRecon on parcellation – Comparison against state-of-the-art 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 graph convolutional network (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 proposed 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

5.4 Discussion and Conclusion

We presented SegRecon, a novel deep learning end-to-end model for the joint reconstruction and segmentation of nested surfaces, directly from MRI volumes. Our model learns multiple signed distance functions that represent surfaces implicitly as iso-levels. An inter-surface distance loss, computed from the distance maps during training, ensures that surfaces do not cross and that the predicted cortical thickness is anatomically possible. After applying a topological correction method (Bazin & Pham, 2007), a mesh is generated for each surface from their signed distance map using the Marching Cubes algorithm (Lorensen & Cline, 1987). Jointly, the model also learns to predict the spherical coordinates of each voxel in a registered atlas space. The propagation of labels from the atlas space effectively segments the cortical white matter surface.

Our experiments used the largest publicly available dataset of manually-labeled brain surfaces (Klein *et al.*, 2017), as well as the ABIDE-I (Di Martino *et al.*, 2014) and OASIS (Marcus *et al.*, 2007) datasets, to evaluate the surface reconstruction and segmentation accuracy of our method. We first showed the advantage of employing a signed distance map over a binary surface mask for reconstructing cortical surfaces. When comparing surfaces reconstructed by our method to those produced by FreeSurfer, using a continuous signed distance map significantly reduces the Hausdorff distance from 2.8 mm to 1 mm. Fig. 5.3 shows the irregularities and artifacts in the reconstructed surface due to the use of binary map. We then validated the robustness of our reconstruction method on the ABIDE and OASIS datasets which were not used in training. The method yields a Hausdorff distance less than 1.2 mm on samples from these datasets, obtained with varying acquisition protocols and corresponding to subjects with very different age and cortical alterations. Surfaces reconstructed by our method, presented in Fig. 5.3 and 5.4, are visually similar to FreeSurfer meshes which require extensive runtimes to generate.

We analyzed the impact on performance of the reference atlas selected for transferring cortical parcellation labels to the surface. While Dice scores ranging from 84.60% to 87.33% were obtained with 5 different atlases, an improved Dice of 88.69% was achieved via a multi-atlas strategy combining the predictions for different atlases with 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 than directly predicting cortical labels with 3D-UNet (79.9%) which, unlike our method, cannot be used to reconstruct cortical surfaces. Moreover, it achieved a significantly higher mean Dice score than FreeSurfer (84.3%) with substantially reduced computation times over compared to this method (hours vs. seconds). Likewise, it improved by over 2% Dice a state-of-art parcellation method based on GCN that requires pre-computed surfaces as input.

In summary, our work proposes a joint reconstruction and segmentation of nested brain surfaces, operating directly on MRI volumes, that can reconstruct and segment cortical surfaces faster and more accurately than existing approaches. While the potential of our method is demonstrated on the cerebral cortex, it could be applied to various other surface data such as cardio-vascular surfaces. Moreover, although our model includes a loss to control the distance between reconstructed surfaces and prevent them from crossing one another, incorporating more powerful topological constraints during network training could remove the need for topological correction in post-processing. Additionally, using a local smoothing based on anisotropic diffusion (Perona & Malik, 1990), instead of Gaussian smoothing, could help regularize the mesh while better preserving cortical folding patterns.

CONCLUSION AND RECOMMENDATIONS

The literature reviewed in introduction, highlights the challenges of surface analysis and the limitation of working directly on brain surfaces. This thesis addresses these challenges by proposing a set of geometric frameworks to directly learn brain surface data. Specifically, the four research objectives led to novel tools useful for fast and accurate brain surface segmentation, classification and reconstruction. In this chapter of the thesis, each contribution for the three objectives are summarized with its practical impact, current limitations and the possible directions for future works discussed.

6.1 Summary of contributions

Objective 1: Spectral graph convolutions for cortical surface parcellation

In Chapter 2, we have proposed a general novel methodology based on spectral graph theory and graph convolution for brain surface analysis. The proposed framework uses advancement in spectral matching to learn across multiple surface domains. The spectral filters of our graph convolutions network learns data to perform cortical surface parcellation. Moving from the conventional Euclidean domain to spectral domain for surface analysis reveals a significant gain in performance on the largest manually labeled MindBoggle dataset. Results also show that our framework leverages the geometric neighborhood information effectively compared to the state-of-the-art approaches. Furthermore, experiments indicate the computation advantages by providing surface parcellation in seconds rather than hours. The benefits of the method is multi-fold, on one hand computationally, and on the other, opening new strategy for geometry aware surface analysis.

Impact: The findings in this chapter has the potential to impact learning surface data directly with applications to multiple neuroscience studies. The proposed spectral graph convolution method overcomes the limitations of current spectral approaches, which are restrained to a fixed graph structure. It also provides an exhaustive evaluation on a multi-centric, publicly available data indicating the robustness of the method.

Objective 2: Learnable pooling in graph convolutional networks for brain surface analysis

Chapter 3, presented a new framework that enables learnable pooling operation on graphs of arbitrary structure. The proposed pooling operation via spectral graph convolutions performs surface classification and regression tasks. The method leverages the spectral embedding of the the graph nodes to learn spatially consistent regions to aggregate features across the convolution layers. Experiments on the MindBoggle dataset of 101 brain surfaces, reveal the correspondence between the learned pooling regions and the final downstream task. Our results on ADNI dataset for predicting the stages in Alzheimer's disease and regressing the brain age show significant improvements in performance over other surface=based approaches. Moreover, a thorough ablation study validates the architectural choices for our method.

Impact: The proposed learnable pooling is, to the best of our knowledge, the first method to perform surface analysis across the dataset with varying size and structure. It offers a task driven pooling regions that could potentially reveal new biomarkers for detecting neurological disorders.

Objective 3: Alignment invariant brain surface analysis Chapter 4, proposes a novel adversarial domain adaptation framework to overcome the limitation of aligning the spectral embedding of brain surfaces. In a data driven approach, this work utilized an adversarial training mechanism to perform alignment independent surface parcellation. Domain invariant surface parcellation is evaluated on manually labeled MindBoggle dataset. The results reported in this chapter show performance improvement for cortical parcellation over spectral graph convolution network without any adversarial training, in a fraction of the time.

Impact: The matching of spectral basis is an essential procedure when learning across surface in spectral domain. However, with the proposed method in the chapter 4, the pre-processing of aligning the spectral embedding of surfaces could be removed. This aids in a direct domain invariant learning of surface data.

Objective 4: Direct joint cortical surface reconstruction and segmentation from MRI volumes

Chapter 5, proposes an end-to-end deep learning model for joint surface reconstruction and segmentation from a MRI volume. Two continuous signed distance maps to the white and pial surfaces are predicted densely as an implicit description of surfaces. Topologically correct cortical meshes are reconstructed with Marching cubes algorithm. Jointly the methods also learn spherical coordinates to the registered atlas to segment the cortical surface. This method overcomes the conventional approach that involve geometric simplification based smoothing and slow segmentation. The performance of the surface reconstruction was evaluated using with experiments on MindBoggle (Klein *et al.*, 2017), ABIDE (Di Martino *et al.*, 2014) and OASIS (Marcus *et al.*, 2007) datasets. The results on unseen test datasets show the robustness of the proposed method. The reconstructed surfaces of our method show qualitatively similar results to FreeSurfer generated meshes while the method is orders of magnitude faster.

Impact: This works contributes to the first study for joint brain surface reconstruction and segmentation from MRI volumes. The method as such can directly be used for extracting and segmenting other organs. The significant reduction in computation time for this joint task could potentially help in targeted drug treatment for diseases manifested on the brain surface.

6.2 Limitations and future work

The contributions of this thesis are discussed in the above section. In spite of this, there remains limitations that were not thoroughly investigated. In this section, the main limitations are identified, and recommendations for future works are provided.

Spectral embedding estimation: The surface analysis, including parcellation and classification proposed in this thesis, is dependent on the spectral decomposition of the graph Laplacian matrix. The assumption made for the spectral methods is that the topology of the graphs or surfaces is equivalent across the dataset. A developed surface analysis method could have an impact on a dataset with varying geometry or having missing parts in the surface mesh. Employing a robust method to evaluate multi-geometry structure could reveal new biomarkers for disease prediction. Additionally, the computation time for spectral decomposition is dependent on the size of the graph, which could burden the overall time complexity of the system. Even though our proposed graph analysis method can compensate for the overall time, in the future, we could focus on using faster decomposition methods (Li, Lian, Kwok & Lu, 2011) or parallel computing with GPUs.

Learning brain graph structure: In this thesis, we proposed learning algorithms capable of estimating statistics for localized anatomical regions. However, handling the longitudinal surface data would be challenging for these frameworks. For instance, temporal/longitudinal prediction of surface atrophy for a subject has varying graph structures across multiple time points. Missing data due to an unsuccessful scan is also a common problem with the longitudinal analysis. Algorithms such as (Song, Zheng, Song & Cui, 2018) can dynamically learn the graph connectivity or adjacency matrix through backpropagation. Adapting these frameworks for temporal studies could potentially aid in finding biomarkers for Alzheimer's disease.

Graph Un-pooling: The learnable graph pooling strategy contribution in this thesis is used for classification and regression tasks. However, the limitation of graph un-pooling operation hinders hierarchical encoder-decoder analysis. This graph encoder-decoder architecture has the potential for segmentation, obtaining unique lower-dimensional representation, and many more applications. Current approaches like Graph Unet (Gao & Ji, 2019) proposes to use fixed pooling un-pooling operations. Similar to the potential of learning pooling operation, learning to un-pool could highlight the anatomical regions revealing new disease biomarkers. Moreover, by incorporating unpooling operations in the proposed model, we could also explore applications requiring node-level outputs like regressing cortical thickness over time.

Uncertainty estimation: Automatic segmentation of cortical surfaces into multiple anatomical or functional regions is proposed in this thesis. Manually labeling the surface is a challenging and time consuming task. Trainable models use registered atlas labels or manual annotations to learn the parcel regions. The boundary for each parcel can be uncertain depending on the expert or the type of labeled atlas (MNI or DKT). Similarly, the learned model depending on the complexity, could have uncertain boundary regions. Recent methods (Dgani, Greenspan & Goldberger, 2018; Zhao, Chen, Hu & Cho, 2020) aim to understand and investigate both aleatoric (data dependent) and epistemic (model dependent) uncertainties when using the graph convolution models. Quantifying uncertainties for specific tasks like parcellation could help clinicians in explaining and interpreting the complex surface data and the learned model.

Reconstruction of surfaces: A contribution of this thesis is proposing a joint reconstruction and segmentation of nested brain surfaces. While the potential of our method is demonstrated on the cerebral cortex, it could be applied to various other surface data such as cardio-vascular surfaces. Depending on the domain, adding a topological or anatomical constraint could extract robust surfaces. Another future contribution could be to include an end-to-end deep learning model for input image normalization and higher resolution synthesis to reconstruct multiple resolution surfaces.

Other surface analysis applications: The potential of our spectral graph convolution is demonstrated on cortical parcellation. It can be applied to other analyses of surface data.

For instance, our framework has a direct impact on other morphometric measurements on surfaces, for instance, regressing cortical thickness or curvature. Likewise, the potential of GCN with pooling operation is shown for classification. This method could be extended to brain fingerprinting or shape based brain retrieval systems. The use of adversarial learning could be helpful in semi-supervised surface segmentation, thereby mitigating the need for large amounts of labeled surfaces. Currently, the spectral graph representation provides a global description of the shapes useful for population based analysis. The impact of higher frequency input representations could be evaluated in future work to embed the local information.

In summary, the findings of the thesis provide new geometric tools for brain surface analysis. The first research objective led to learning algorithms capable of estimating statistics for localized anatomical regions with high speed and accuracy. The subsequent research objective led to the development of a learnable pooling operation for graph convolution networks. The proposed approach was found to be useful to brain surface analysis, but general enough for other applications. The final objective enables fast and accurate joint brain surface reconstruction and segmentation from MRI volumes. These works proposed in the thesis with concrete recommendations for future work will significantly assist clinicians in the early prediction of cortical atrophy and planning treatments for diseases.

APPENDIX I

SPECTRAL GRAPH TRANSFORMER NETWORKS FOR BRAIN SURFACE PARCELLATION

Karthik Gopinath[a], He Ran[b], Christian Desrosiers[c], Herve Lombaert[d]

[a,b,c,d] Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

Paper published in *IEEE International Symposium on Biomedical Imaging* (ISBI), April 2020

Presentation

This chapter presents the article *"Graph Domain Adaptation for Alignment-Invariant Brain Surface Segmentation"* (He *et al.*, 2020) submitted to **ISBI** conference (IEEE International Symposium on Biomedical Imaging), published on April 2020. The article was presented as an oral talk, held virtually at this conference. The first two authors have contributed equally to this work. The objective of this article is learn the transformation matrix for aligning brain surfaces in spectral domain.

A.I.2. Introduction

The surface of a human brain is a complex geometrical structure containing multiple convoluted folding patterns. Statistical analysis of the brain surface aids in understanding its anatomy, and machine learning methods are often sought for automating this analysis. Conventional machine learning frameworks exploit spatial information from the Euclidean domain such as image or volumetric coordinates Hua *et al.* (2013); Zhang & Davatzikos (2011). Similarly, state-of-the-art deep learning approaches Dolz *et al.* (2017); Kamnitsas *et al.* (2017b) operate on data lying in Euclidean spaces, offering a drastic speed advantage over traditional methods. However, the geometry of the brain is highly variable, hindering the direct use of these modern deep learning algorithms over multiple brain surfaces.

Recently, deep learning approaches on irregular graphs Bronstein *et al.* (2017); Levie *et al.* (2018); Monti *et al.* (2017) have been proposed. These methods formulate a convolution theorem from Fourier space to spectral domains over graphs. One main limitation of these spectral approaches is their lack of expressing surface data in comparable spectral bases across different surface domains Bronstein *et al.* (2013); Eynard *et al.* (2015); Kovnatsky *et al.* (2013). The Laplacian eigenbases are indeed incompatible across multiple geometries, challenging their direct use during training. As a solution, some recent work Boscaini *et al.* (2016); Masci *et al.* (2015) maps the local information onto geodesic patches and uses conventional template matching in spatial convolutions. For instance, Monti *et al.* (2017) proposed local convolution operation as filtering over small neighborhoods in spatial domain. Their spatial representations of surface data remain, however, defined in a Euclidean space by using polar representations of pixels or mesh vertices.

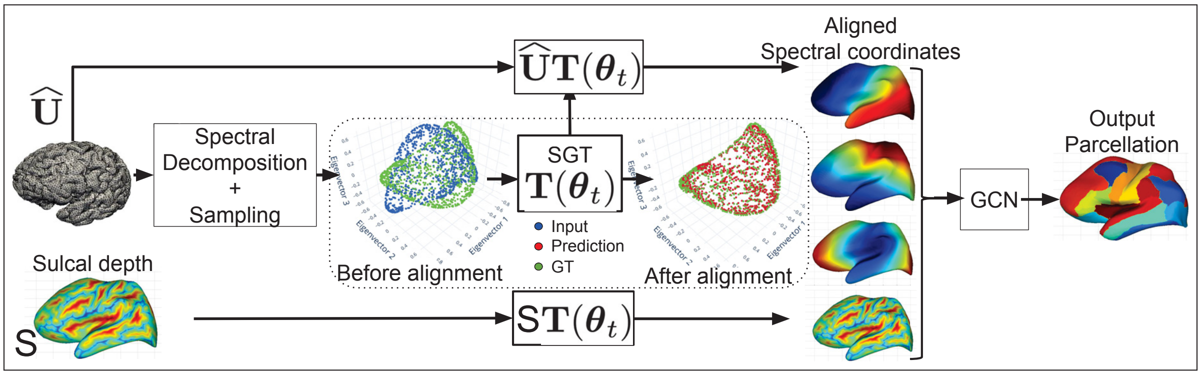


Figure A.I.1 **Overview of the our spectral graph transformer architecture** – The spectral decomposition of the brain graph is randomly sub-sampled as an input point cloud to a SGT network. The SGT learns the transformation parameters aligning the eigenvectors of multiple brains. The transformation matrix is multiplied with original spectral coordinates to feed the GCN for parcellation. The point cloud is illustrated before and after alignment with our SGT network. The GCN architecture follows recommendations from Gopinath *et al.* (2019b).

In the literature, spectral graph matching has been used to transfer surface data across aligned spectral domains Lombaert *et al.* (2015a). Such strategy Gopinath *et al.* (2019b) enables the learning of spectral graph convolution networks across multiple surface data. These methods, however, involve an explicit computation of a transformation map for each brain towards one reference template. This process of aligning the eigenvectors of graph Laplacians is currently an important computational bottleneck. This expensive step is necessary in such approach to handle the differences across eigenvectors, including sign flips, ordering, and mixing of eigenvectors in higher frequencies. In this work, we propose a framework for learning this transformation function across multiple brain surfaces. In an alternative application for natural image classification, Jaderberg, Simonyan, Zisserman et al. (2015) proposes a transformer network for CNNs for learning a transformation matrix to spatially standardize the image data. Similarly, Qi, Su, Mo & Guibas (2017a) also proposes a transformation network for learning over point clouds of geometric structures. These methods are, however, limited to pointwise information in a Euclidean space. This paper introduces a Spectral Graph Transformer Network (SGT) to learn the parameters for aligning multiple surfaces directly in the spectral domain. We illustrate the learning capabilities of this approach with an application to brain parcellation. We use the aligned coordinates from our SGT network along with a graph convolution network (GCN) for quantifying parcellation. The learnt alignment of 101 manually-labeled brain surfaces Klein *et al.* (2017) reveals that our approach improves brain parcellation by 4.4%, from an average Dice overlap of 78.8% to 83.2%. The performance of our method is shown to be at par with traditional alignment strategies, performing at 84.4%, but gains a significant speed improvement. The learning of an end-to-end SGT and GCN model enables a direct, automatic learning of surface data across multiple brains. Our SGT part learns a transformation matrix that handles the eigenvector differences, while the GCN part focuses on the brain parcellation. The next section details the fundamentals of our SGT and GCN model, followed by an evaluation of our alignment strategy for graph convolutions.

A.I.3. Method

An overview of the method is shown in Fig. A.I.1. Firstly, the cortical surfaces modeled as brain graphs are embedded in a spectral manifold using the graph Laplacian operator. Secondly, graph nodes are randomly sampled in the spectral embeddings and fed to the SGT network to align the brain embeddings. Finally, a GCN provides a labeled graph as output, taking spectral coordinates and cortical sulcal depth as input.

A.I.3.1 Spectral embedding of brain graphs

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a brain graph defined with node set $\mathcal{V}$, such that $|\mathcal{V}| = N$, and edge set $\mathcal{E}$. Each node i has a feature vector $\mathbf{x}_i \in \mathbb{R}^3$ representing its 3D coordinates. We map $\mathcal{G}$ to a low-dimension manifold using the normalized graph Laplacian operator $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{A}$ is the weighted adjacency matrix and $\mathbf{D}$ the diagonal degree matrix. In this work, we define the weight between two adjacent nodes as the inverse of their Euclidean distance. Let $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ be the eigendecomposition of $\mathbf{L}$, the normalized spectral coordinates of nodes are given by $\widehat{\mathbf{U}} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}$.

A.I.3.2 Spectral transformer network

The normalized spectral coordinates $\widehat{\mathbf{U}}$ from the spectral embedding of $\mathbf{L}$ is only defined up to an orthogonal transformation. We thus need to align the spectral representations of different brain graphs to a common representation. As a base reference, we align the normalized spectral embedding of all the brain surfaces to a template $\widehat{\mathbf{U}}_{\text{ref}}$ in the dataset. This traditional alignment process involves computing an expensive optimal orthogonal transform based on iterative Procustes algorithm Lombaert *et al.* (2015a), which can be formulated as

$$\arg\min_{\pi,\mathbf{T}} \sum_{i=1}^{N} \left\|\mathbf{T}\,\widehat{\mathbf{u}}_i - \widehat{\mathbf{u}}^{(0)}_{\pi(i)}\right\|_2^2 \tag{A I-1}$$

This alignment step is computationally expensive, taking few seconds to converge. Also, the alignment process is independent of the final target task. Our STN consists of learning the transformation matrix $\mathbf{T}$ for every brain graph in a data-driven manner. As input to the network, we provide $\widehat{\mathbf{U}}_{\text{sub}}$, a set of N randomly sub-sampled $\widehat{\mathbf{U}}$ coordinates, chosen similarly to Qi *et al.* (2017a), with enough samples to recover $\mathbf{T}$. Since most information on graph connectivity is encoded in the first eigencomponents of $\mathbf{L}$, to limit processing times, we only keep the first 3 components for the learning step. Thus, $\widehat{\mathbf{U}}_{\text{sub}}$ is a matrix of size $N{\times}3$. Fig. A.I.1 describes the architecture of our spatial transformer network.

The model first applies a sequence of two point-wise linear transformation layers on $\widehat{\mathbf{U}}_{\text{sub}}$, each one followed by a non-linear rectifier (ReLU) function. Such layer takes a $N{\times}M_{l-1}$ matrix $\mathbf{X}$ as input and post-multiplies it by a $M_{l-1}{\times}M_l$ parameter matrix $\mathbf{W}_l$ to give an output matrix of size $N{\times}M_l$. This transformation, which is similar to 1×1 convolutions in CNNs, expresses each embedded node with respect to a shared set of M_l hyper-planes in the spectral space, and is used to capture the global shape of the embedding. In our model, we use $M_1 = 256$ for the first layer and $M_2 = 128$ for the second one (note that $M_0 = 3$). Next, the output of the second point-wise transformation layer is converted to a fixed-size representation of size 128×1 by applying average pooling. Last, to get the final spectral transformation matrix, we apply three MLP layers of size [128, 64, 9], also with ReLU activations, and reshape the output of the last layer into a 3×3 matrix. This transformation matrix is multiplied to the normalized spectral coordinates $\widehat{\mathbf{U}}$ to obtain the aligned spectral coordinates.

The parameters of the spectral transformer network are optimized by computing the mean square error between the predicted coordinates and spectral coordinates $\widetilde{\mathbf{U}}$ obtained with the iterative alignment method. To enforce regularization during training, and match the possible rotation and flip ambiguity of eigendecomposition, we also add a second loss term imposing the transformation matrix to be orthogonal. The final loss function is given by

$$E_{\text{spt}}(\mathbf{\Theta}_t) \;=\; \|\widetilde{\mathbf{U}} - \widehat{\mathbf{U}}\,\mathbf{T}(\mathbf{\Theta}_t)\|_F^2 \;+\; \|\mathbf{T}(\mathbf{\Theta}_t)\mathbf{T}^\top(\mathbf{\Theta}_t) - \mathbf{I}\|_F^2 \qquad \text{(A I-2)}$$

A.I.3.3 Graph convolution on surfaces

The second part of our end-to-end model is based on a geometric convolutional neural network that maps the now-aligned spectral coordinates to a common comparable graph embedding. A generalized convolution operation on a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, with $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$, as the neighbors of node $i \in \mathcal{V}$, is defined as

$$z_{ip}^{(l)} = \sum_{j \in \mathcal{N}_i} \sum_{q=1}^{M_l} \sum_{k=1}^{K_l} w_{pqk}^{(l)} \cdot y_{jq}^{(l)} \cdot \varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j; \mathbf{\Theta}_k^{(l)}) + b_p^{(l)}, \qquad \text{(A I-3)}$$

where $\varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j; \mathbf{\Theta}_k)$ is a symmetric kernel in the embedding space with parameter $\mathbf{\Theta}_k$. In this work, we follow Gopinath *et al.* (2019b) and use a Gaussian kernel: $\varphi(\widehat{\mathbf{u}}_i, \widehat{\mathbf{u}}_j; \boldsymbol{\mu}_k, \sigma_k) = \exp\big(-\sigma_k \, \|(\widehat{\mathbf{u}}_j - \widehat{\mathbf{u}}_i) - \boldsymbol{\mu}_k\|^2\big)$.

We define a fully-convolutional network comprising of 4 graph convolution layers with sizes 256, 128, 64, and 32. Each layer have $K_l = 6$ Gaussian kernels similar to Gopinath *et al.* (2019b). The total target parcels are 32, hence, our last layer is of size 32. Leaky ReLU is applied after each layer to obtain our filter responses. A softmax operation is used after the last graph convolution layer in order to obtain the probabilities of the mutually-exclusive parcels at each node. Our output loss function employs a cross-entropy with Dice loss for all parcellations. Our final end-to-end model comprising of a spectral transformer and a graph convolution network for brain parcellation is trained using the loss function given by

$$E_{\text{final}}(\mathbf{\Theta}_t, \mathbf{\Theta}_g) = \lambda \, E_{\text{spt}}(\mathbf{\Theta}_t) + E_{\text{gcn}}(\mathbf{\Theta}_g). \qquad \text{(A I-4)}$$

This final loss E_{final} is minimized by back-propagating the error using standard gradient descent optimization.

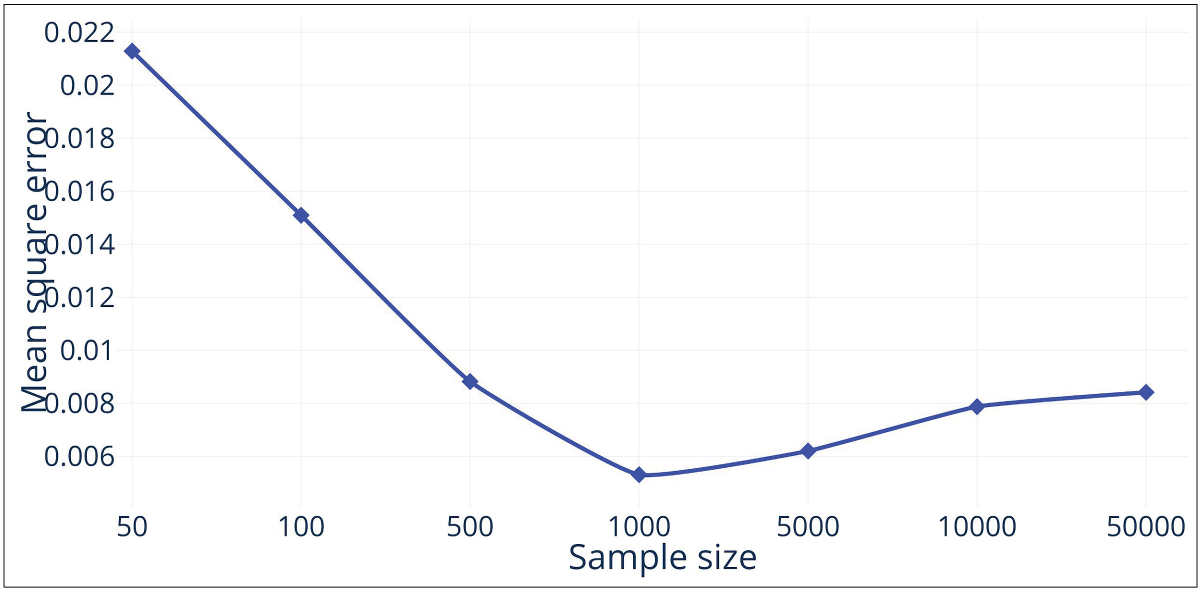


Figure-A I-1 **SGT network data sampling vs. mean square error –** Each point indicates the performance of SGT model in terms of mean square error. It is observed that the model trained with fewer nodes than 500 perform poorly compared to all the models. The best mean square error is achieved for model with 1000 nodes as input to SGT.

A.I.4. Experiments and Results

In this section, we evaluate how inputs affect our SGT network. The optimal SGT parameters are thereafter used to train our end-to-end model for brain parcellation. We validate our approach on the Mindboggle Klein *et al.* (2017) dataset containing manually-labeled brain surfaces. The dataset contains 101 cortical meshes, each with 102K to 185K vertices and 32 manually-labeled parcels. We randomly split the dataset into training, validation and testing in a 70-10-20% ratio for our experiments. Here, we induce random sign flips on the eigenvectors of the training dataset to balance flipping and rotation variance. The performance of the methods are measured in terms of average Dice overlap and Hausdorff distances. The experiments are carried out on an i7 desktop computer with 16GB of RAM and a Nvidia Titan X

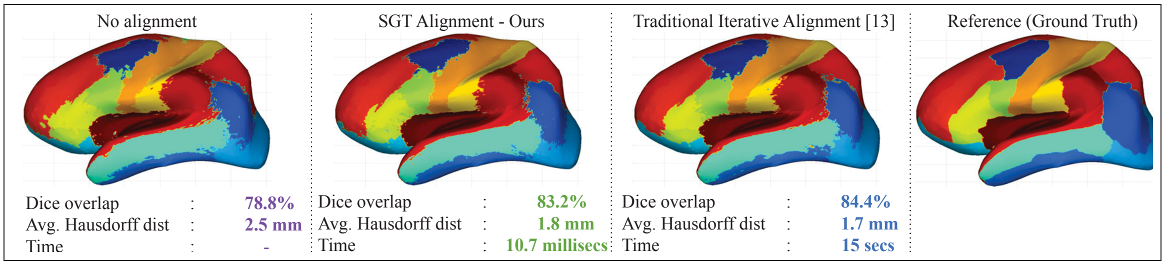


Figure A.I.2 **Brain parcellation** – Performance comparison of different alignment strategies using GCN measured with average Dice overlap and Hausdorff distance. Model trained with no SGT yields low Dice of 78.6% with irregular segmentation boundaries. Training an end-to-end SGT and GCN model achieves a Dice overlap of 83.2% similar to the performance of a traditional alignment model with Dice overlap of 84.4%. The Hausdorff distance and qualitative results show very similar results between the two methods. However, a significant speed gain in order of 10.7 *milliseconds* is achieved with our SGT and GCN model.

A.I.4.1 Spectral transform data sampling

Our spectral transformer network takes as input a set of points in the spectral domain. The number of eigenvectors is fixed to three, as suggested in Gopinath *et al.* (2019b). To evaluate the effect of input size N, we sample spectral points randomly from 50 to $50,000$. We study the performance of spectral alignment using our SGT model in terms of mean square error.

The results shown in Fig. I-1 illustrate that the best alignment performance is achieved with a sub-sampling size of $N = 1000$. The input data with $N = 50, 100, 500$ is inadequate to capture the complete geometric information of the brain, as seen in Fig. I-1. In addition to lower performance, a higher number of nodes also increases memory consumption and computation time. The gain in mean square error for input size over $N = 1000$ can also be seen in Fig. I-1.

A.I.4.2 Brain surface parcellation

We now evaluate the performance of our end-to-end SGT and GCN model on brain surface parcellation. The predicted transformation matrix from SGT aligns all brain surfaces. The

Table A.I.1 **Different alignment strategies with GCN approaches** – Average Dice overlaps (in %) over 32 parcels on test set are shown along with classification accuracy (in %), and average Hausdorff distances (in millimeters).

Method	**Dice overlap (%)**	**Accuracy (%)**	**Avg. Hausdorff (mm)**
No Alignment	78.82 ± 4.02	81.68 ± 3.88	2.54 ± 2.86
Pretrained + Orthogonal	81.97 ± 3.20	84.14 ± 2.88	1.99 ± 2.19
Pretrained + MSE	82.29 ± 4.46	84.38 ± 4.09	1.94 ± 2.23
End-to-end (Ours - 10.7 *milliseconds*)	83.26 ± 3.66	85.17 ± 3.48	1.85 ± 2.04
Traditional Alignment (Gopinath *et al.*, 2019b) (15 seconds)	84.42 ± 2.59	85.99 ± 2.53	1.76 ± 1.75

number of embedded node coordinates used during training SGT is set to $N = 1000$. These nodes are randomly sub-sampled for each subject during the training of our end-to-end model.

Our method is compared with different alignment strategies for graph parcellation. We show the limitations of ignoring the spectral alignment. The GCN trained with non-aligned spectral coordinates achieves a Dice overlap performance of 78.8%. This low accuracy is due to the incompatibility of eigenbases across brain surfaces. Training our end-to-end SGT with GCN provides a performance improvement of 4.4% for parcellation over no alignment. Next, our transformer network is trained independently from the parcellation task in order to learn the SGT weights. The rationale of this experiment is to evaluate the use of a fixed alignment strategy for learning the GCN model. We evaluate the use of both SGT loss and orthogonal regularization independently. The model trained with only orthogonal regularization has a performance gain of 3.1% from 78.8% to 81.9%. This increase indicates the usefulness of regularization to learn rotation and flipping. We see a further performance boost by training our SGT model with mean square error. Table A.I.1 shows a similar performance gain of 3.4% compared to not using alignment. Note that updating the weights of both SGT and GCN in an end-to-end framework further guides the learning of the transformation matrix. This experiment setup trains the SGT model to learn a transformation most suitable for the parcellation task. Our end-to-end model indeed yields an improvement in average Dice overlap of 83.4% compared to 82.2% when trained separately. The results of the experiments are reported in Table A.I.1.

A.I.5. Discussion and Conclusion

This paper presented a novel end-to-end framework for learning a spectral transformation required for graph convolution networks. The proposed SGT network learns a transformation in the spectral domain that maps input spectral coordinates to a reference set. We first evaluate the optimal size of the coordinate set necessary for training the SGT network. Next, our experiments on brain surface parcellation validate the benefits of our alignment strategy. Training a GCN model without any alignment results in a low Dice overlap and irregular parcel boundaries as shown in Fig. A.I.2. The conventional procedure of aligning different brain surfaces to a reference is an expensive computational step. Our method learns this alignment step automatically by capturing the geometry of the brain, yielding a Dice overlap of 83.2%. Qualitatively, as illustrated in Fig. A.I.2, the performance of our method is similar to a GCN trained with traditional alignment, however computation times are reduced by a 1400-fold, from 15 seconds to 10.7 *milliseconds*. The use of SGT is evaluated in this paper with brain surface parcellation as an application. Nevertheless, our method can potentially be used for other surface analysis problems such as disease classification or identifying new geometry-related biomarkers.

APPENDIX II

CODE AVAILABILITY

- Matlab scripts for Spectral alignment of brain surfaces available at:
  https://github.com/kharitz/spectral_alignment.git

- PyTorch scripts for Spectral graph convolution for cortical parcellation are available at:
  https://github.com/kharitz/spectral_alignment.git

- PyTorch scripts for the Learnable pooling for brain surface analysis are available at:
  https://github.com/kharitz/learnpool.git

APPENDIX III

COPYRIGHT PERMISSIONS FOR FIGURES USED IN LITERATURE REVIEW

- **Copyright permissions/License for Figure 0.1**:

  “*This file is licensed under the Creative Commons Attribution 4.0 Unported license.*”

- **Copyright permissions/License for Figure 0.2 – Left**:

  “*This file is licensed under the Creative Commons Attribution 3.0 Unported license*”

- **Copyright permissions/License for Figure 0.2 – Right**:

  “*This file is licensed under the Wikimedia Commons, the free media repository*”

- **Copyright permissions/License for Figure 0.3**:

  “*This file is licensed under the Creative Commons Attribution 4.0 International license*”

- **Copyright permissions/License for Figure 0.4 – Left**:

  “*This file is licensed under the Creative Commons Attribution 3.0 Unported license*”

- **Copyright permissions/License for Figure 0.4 – Right**:

  “*This file is licensed under the Wikimedia Commons, the free media repository*”

- **Copyright permissions/License for Figure 0.5**:

  “*This file is licensed under the Creative Commons Attribution 4.0 International license*”

- **Copyright permissions/License for Figure 0.6**:

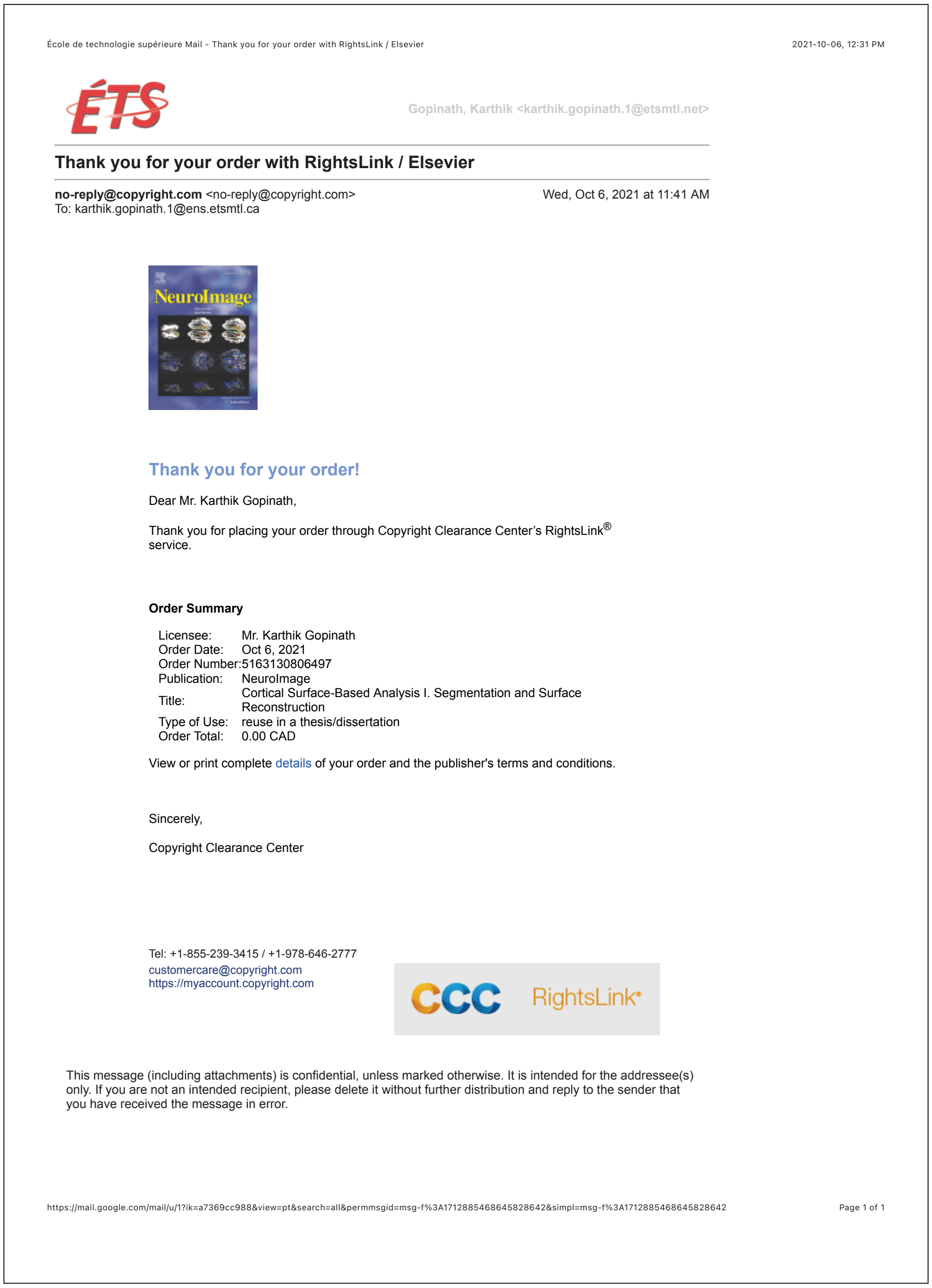
  “*This file is licensed under the Creative Commons Attribution 4.0 International license*”

- **Copyright permissions/License for Figure 1.1**:

  “*This file is licensed under the Creative Commons Attribution 3.0 International license*”

- **Copyright permissions/License for Figure 1.2**:

  “*This file is licensed under the Creative Commons Attribution 3.0 International license*”



École de technologie supérieure Mail - Thank you for your order with RightsLink / Elsevier 2021-10-06, 12:31 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Elsevier

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 11:41 AM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163130806497
Publication:	NeuroImage
Title:	Cortical Surface-Based Analysis I. Segmentation and Surface Reconstruction
Type of Use:	reuse in a thesis/dissertation
Order Total:	0.00 CAD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

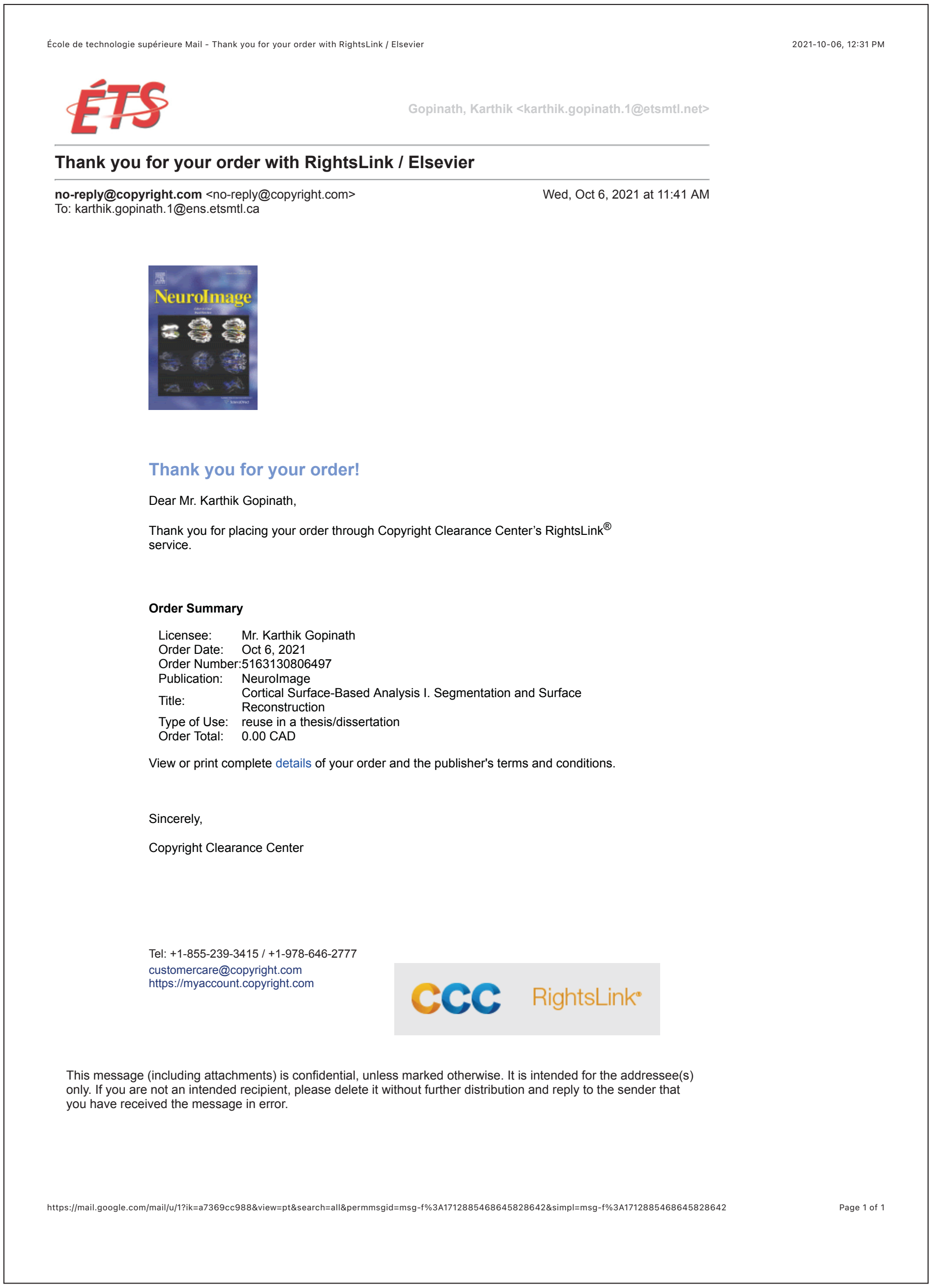
Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

This message (including attachments) is confidential, unless marked otherwise. It is intended for the addressee(s) only. If you are not an intended recipient, please delete it without further distribution and reply to the sender that you have received the message in error.

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&permmsgid=msg-f%3A1712885468645828642&simpl=msg-f%3A1712885468645828642 Page 1 of 1

Figure-A III-1 Copyright permissions for Figure 1.3

- **Copyright permissions/License for Figure 1.3**: see Fig. III-1
- **Copyright permissions/License for Figure 1.4**: see Fig. III-2
- **Copyright permissions/License for Figure 1.5**: see Fig. III-3



École de technologie supérieure Mail - Thank you for your order with RightsLink / Elsevier 2021-10-06, 12:31 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Elsevier

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 11:41 AM
To: karthik.gopinath.1@ens.etsmtl.ca

NeuroImage

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163130806497
Publication:	NeuroImage
Title:	Cortical Surface-Based Analysis I. Segmentation and Surface Reconstruction
Type of Use:	reuse in a thesis/dissertation
Order Total:	0.00 CAD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

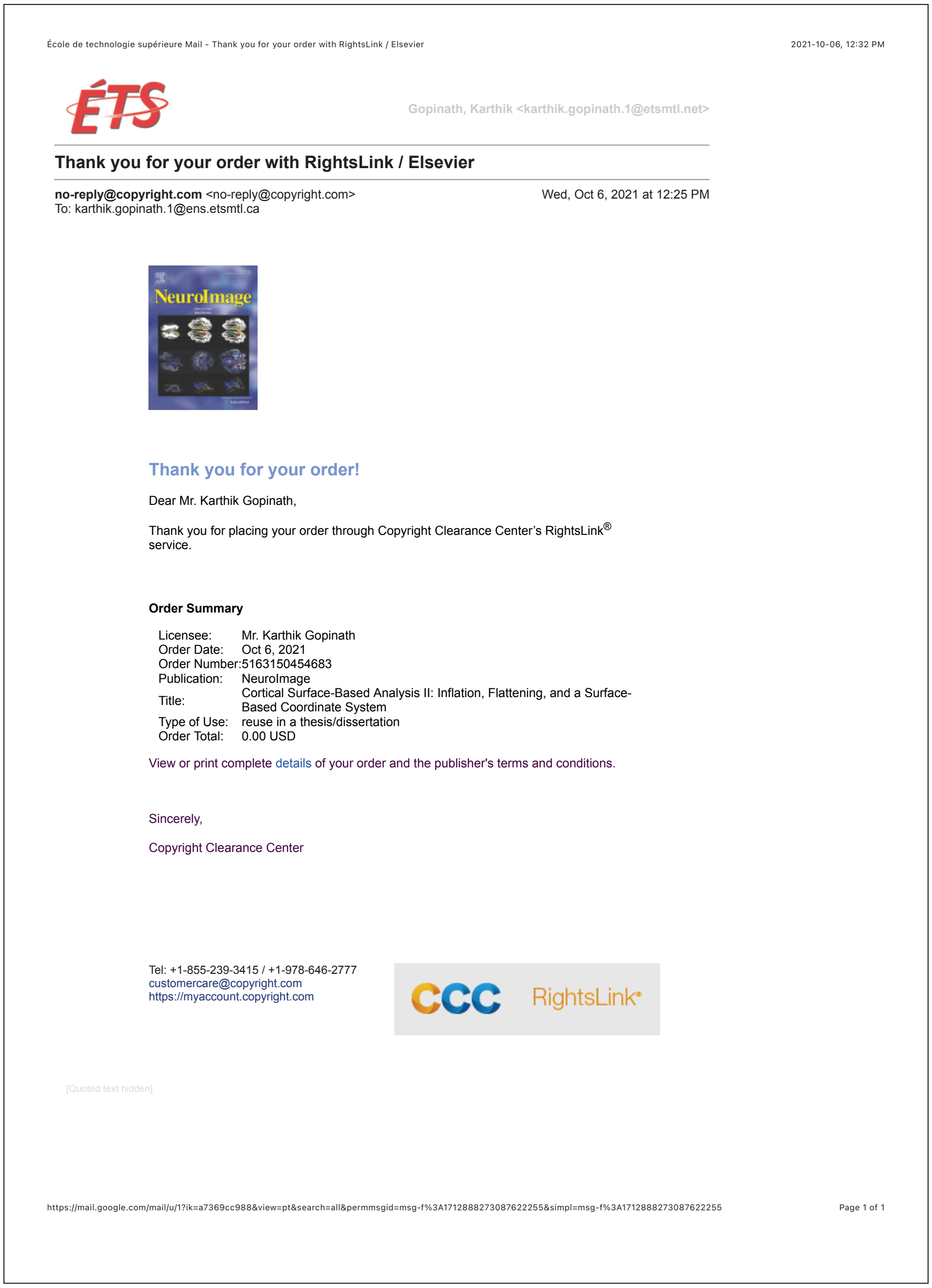
CCC RightsLink®

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&permmsgid=msg-f%3A1712885468645828642&simpl=msg-f%3A1712885468645828642 Page 1 of 1

Figure-A III-2 Copyright permissions for Figure 1.4

- **Copyright permissions/License for Figure 1.6**: see Fig. III-4

- **Copyright permissions/License for Figure 1.7**: see Fig. III-5



École de technologie supérieure Mail - Thank you for your order with RightsLink / Elsevier 2021-10-06, 12:32 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Elsevier

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 12:25 PM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163150454683
Publication:	NeuroImage
Title:	Cortical Surface-Based Analysis II: Inflation, Flattening, and a Surface-Based Coordinate System
Type of Use:	reuse in a thesis/dissertation
Order Total:	0.00 USD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

[Quoted text hidden]

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&permmsgid=msg-f%3A1712888273087622255&simpl=msg-f%3A1712888273087622255 Page 1 of 1

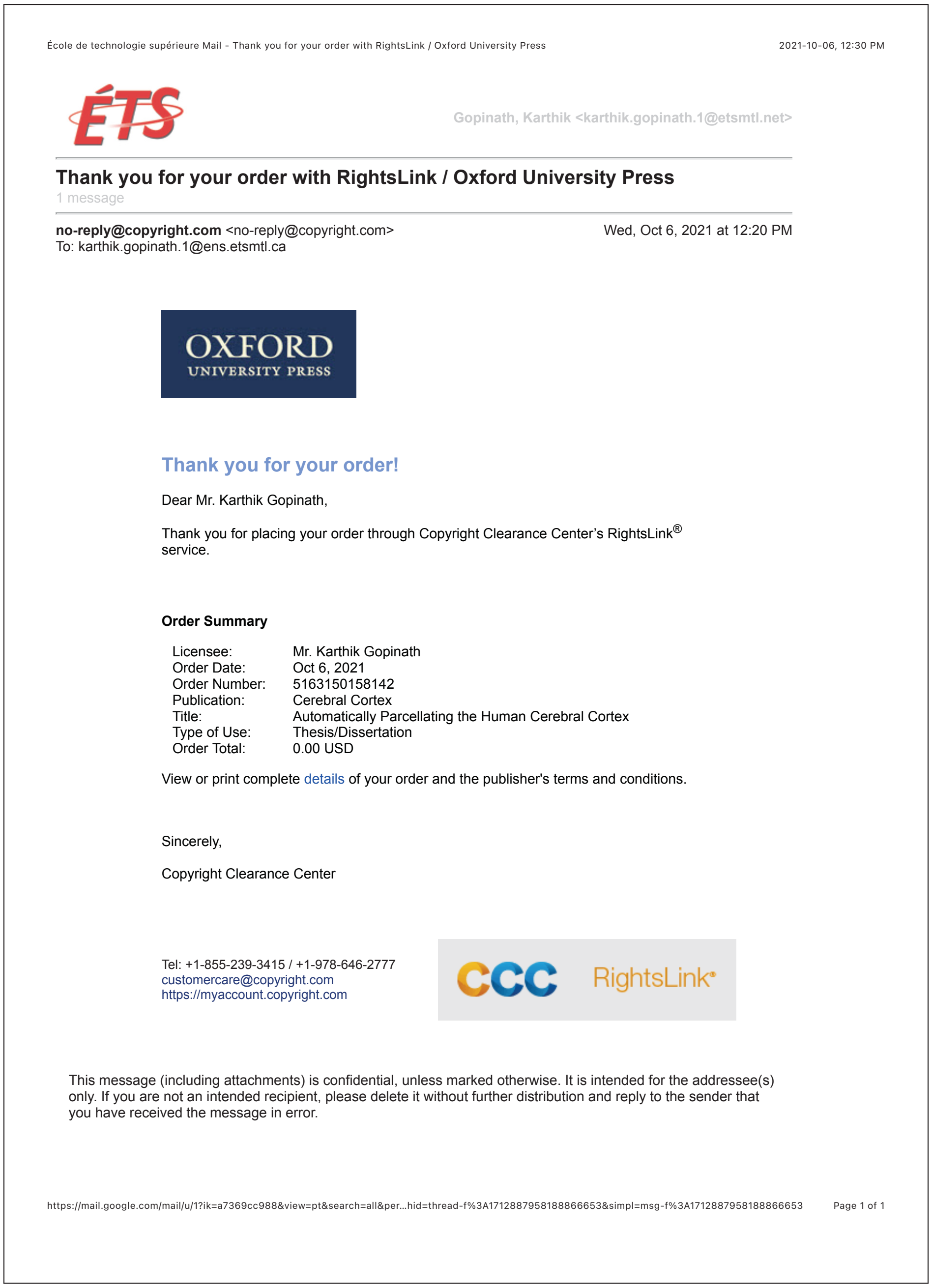
Figure-A III-3 Copyright permissions for Figure 1.5

- **Copyright permissions/License for Figure 1.8**:

  "This file is licensed under the Creative Commons Attribution 4.0 International license"

- **Copyright permissions/License for Figure 1.9**:

  "This file is licensed under the Creative Commons Attribution 4.0 International license"



École de technologie supérieure Mail - Thank you for your order with RightsLink / Oxford University Press 2021-10-06, 12:30 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Oxford University Press
1 message

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 12:20 PM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163150158142
Publication:	Cerebral Cortex
Title:	Automatically Parcellating the Human Cerebral Cortex
Type of Use:	Thesis/Dissertation
Order Total:	0.00 USD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

This message (including attachments) is confidential, unless marked otherwise. It is intended for the addressee(s) only. If you are not an intended recipient, please delete it without further distribution and reply to the sender that you have received the message in error.

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&per...hid=thread-f%3A1712887958188866653&simpl=msg-f%3A1712887958188866653 Page 1 of 1

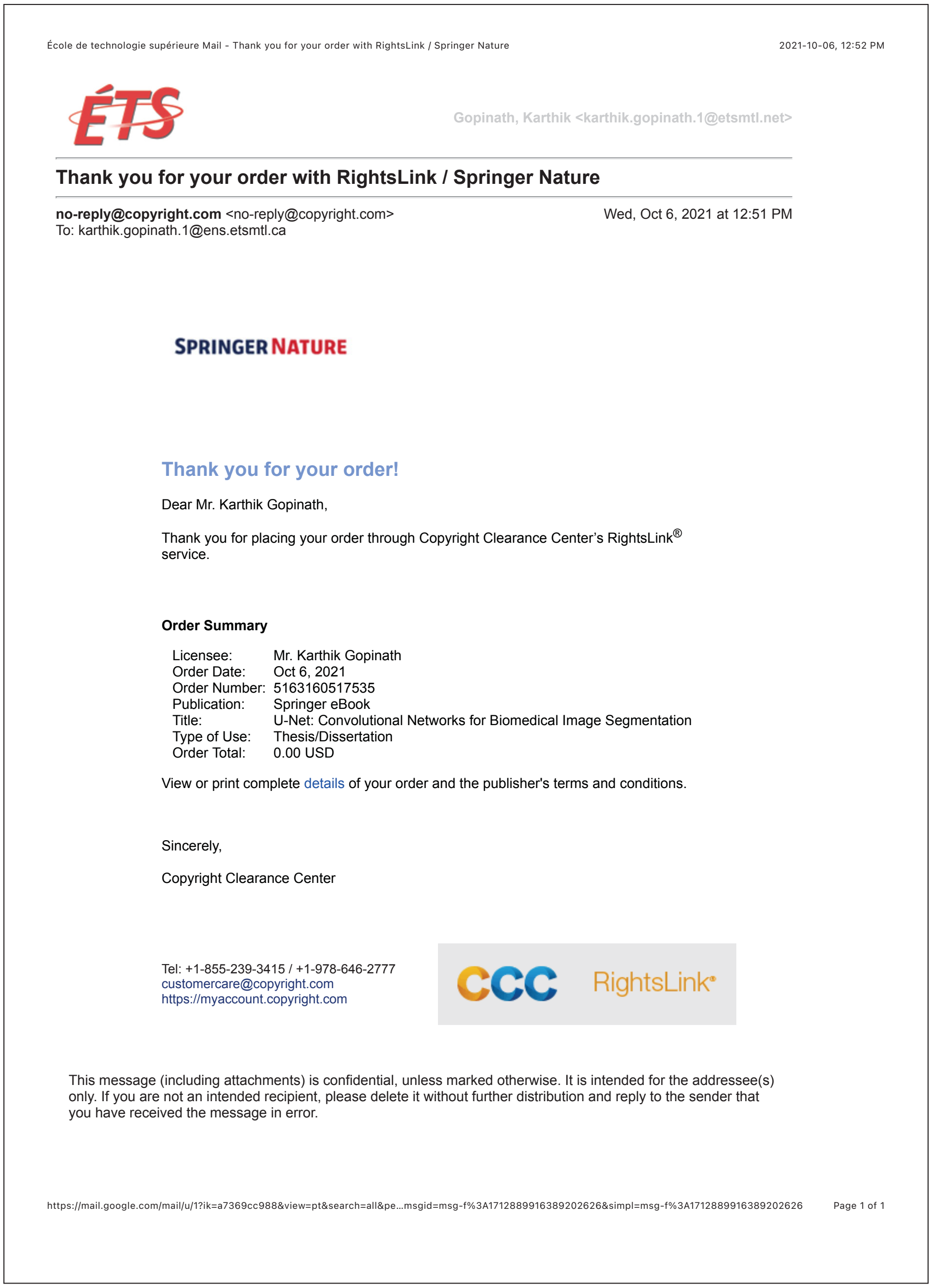
Figure-A III-4 Copyright permissions for Figure 1.6

- **Copyright permissions/License for Figure 1.10**:

  "*This file is licensed under the Creative Commons Attribution 4.0 International license*"

- **Copyright permissions/License for Figure 1.11 – Left**:

  "*This file is licensed under the Creative Commons Attribution 4.0 International license*"



École de technologie supérieure Mail - Thank you for your order with RightsLink / Springer Nature 2021-10-06, 12:52 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Springer Nature

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 12:51 PM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163160517535
Publication:	Springer eBook
Title:	U-Net: Convolutional Networks for Biomedical Image Segmentation
Type of Use:	Thesis/Dissertation
Order Total:	0.00 USD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

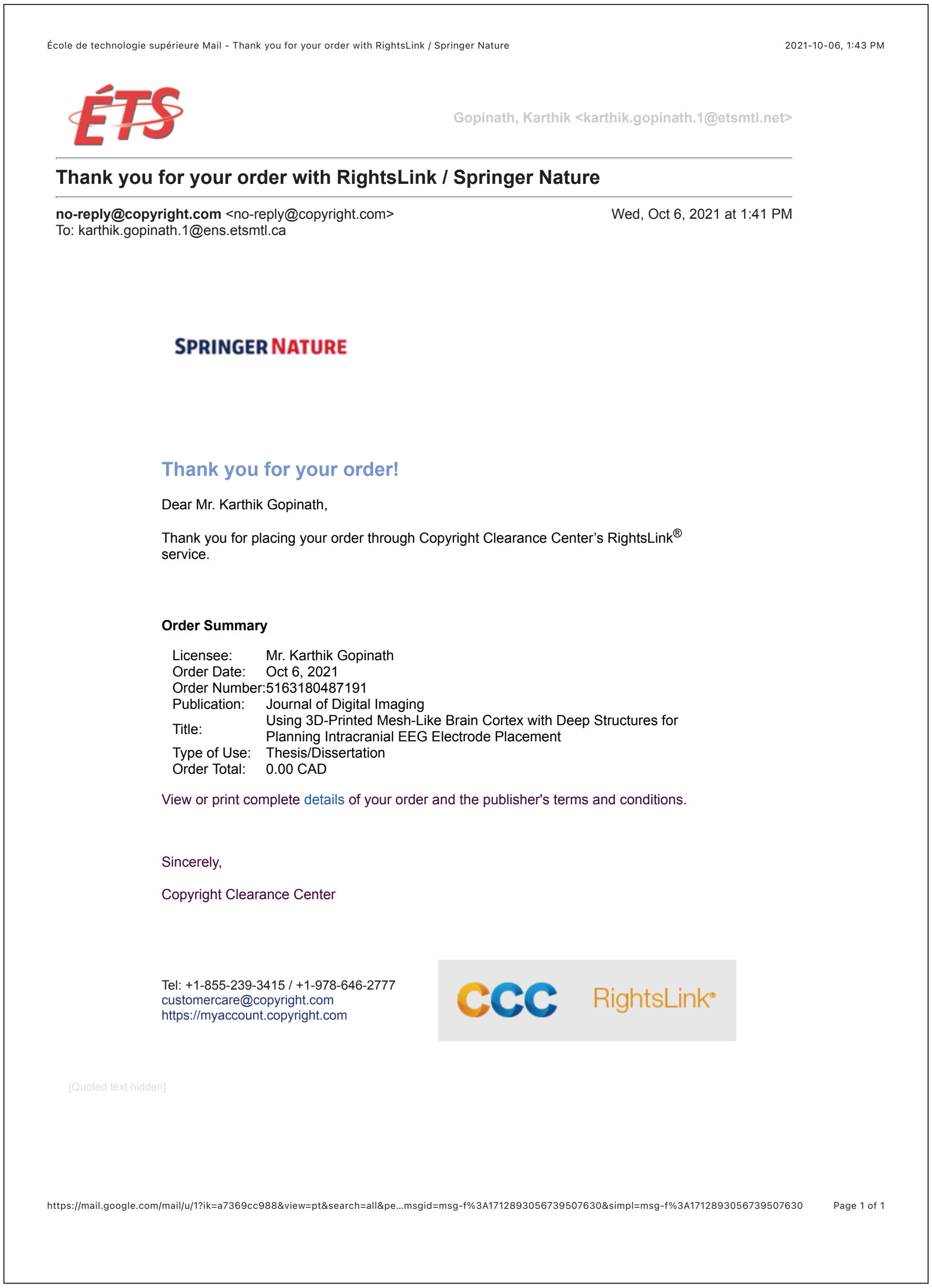
This message (including attachments) is confidential, unless marked otherwise. It is intended for the addressee(s) only. If you are not an intended recipient, please delete it without further distribution and reply to the sender that you have received the message in error.

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&pe…msgid=msg-f%3A1712889916389202626&simpl=msg-f%3A1712889916389202626 Page 1 of 1

Figure-A III-5 Copyright permissions for Figure 1.7

- **Copyright permissions/License for Figure 1.11 – Right** see Fig. III-6:

- **Copyright permissions/License for Figure 1.12** see Fig. III-7:



École de technologie supérieure Mail - Thank you for your order with RightsLink / Springer Nature 2021-10-06, 1:43 PM

ÉTS

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Springer Nature

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 1:41 PM
To: karthik.gopinath.1@ens.etsmtl.ca

SPRINGER NATURE

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163180487191
Publication:	Journal of Digital Imaging
Title:	Using 3D-Printed Mesh-Like Brain Cortex with Deep Structures for Planning Intracranial EEG Electrode Placement
Type of Use:	Thesis/Dissertation
Order Total:	0.00 CAD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

CCC RightsLink®

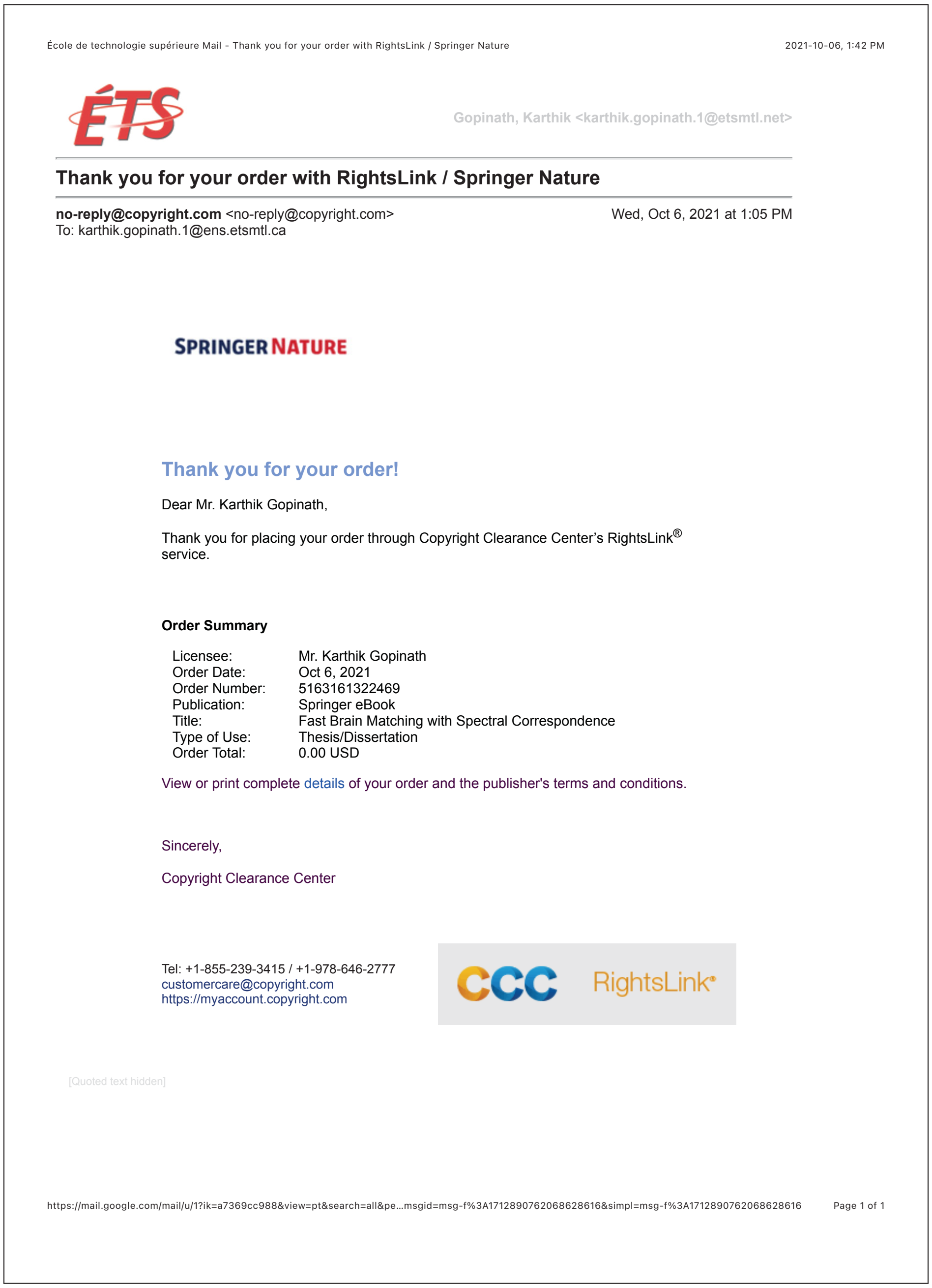
[Quoted text hidden]

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&pe…msgid=msg-f%3A1712893056739507630&simpl=msg-f%3A1712893056739507630 Page 1 of 1

Figure-A III-6 Copyright permissions for Figure 1.11

- **Copyright permissions/License for Figure 1.15** see Fig. III-8:

- **Copyright permissions/License for Figure 1.17** see Fig. III-9:



École de technologie supérieure Mail - Thank you for your order with RightsLink / Springer Nature 2021-10-06, 1:42 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Springer Nature

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 1:05 PM
To: karthik.gopinath.1@ens.etsmtl.ca

SPRINGER NATURE

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163161322469
Publication:	Springer eBook
Title:	Fast Brain Matching with Spectral Correspondence
Type of Use:	Thesis/Dissertation
Order Total:	0.00 USD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

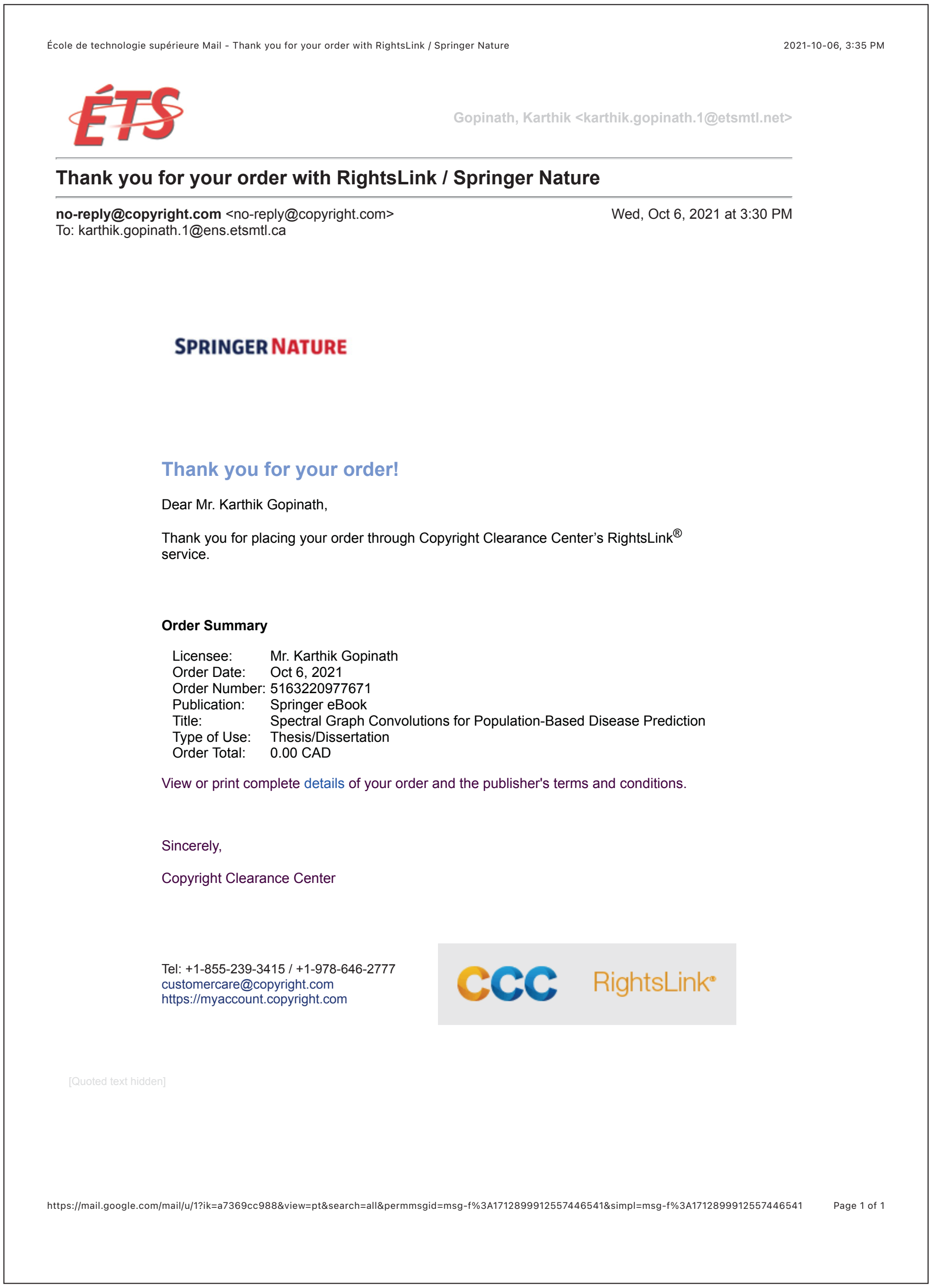
Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

CCC RightsLink®

[Quoted text hidden]

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&pe…msgid=msg-f%3A1712890762068628616&simpl=msg-f%3A1712890762068628616 Page 1 of 1

Figure-A III-7 Copyright permissions for Figure 1.12



École de technologie supérieure Mail - Thank you for your order with RightsLink / Springer Nature 2021-10-06, 3:35 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Springer Nature

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 3:30 PM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee: Mr. Karthik Gopinath
Order Date: Oct 6, 2021
Order Number: 5163220977671
Publication: Springer eBook
Title: Spectral Graph Convolutions for Population-Based Disease Prediction
Type of Use: Thesis/Dissertation
Order Total: 0.00 CAD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

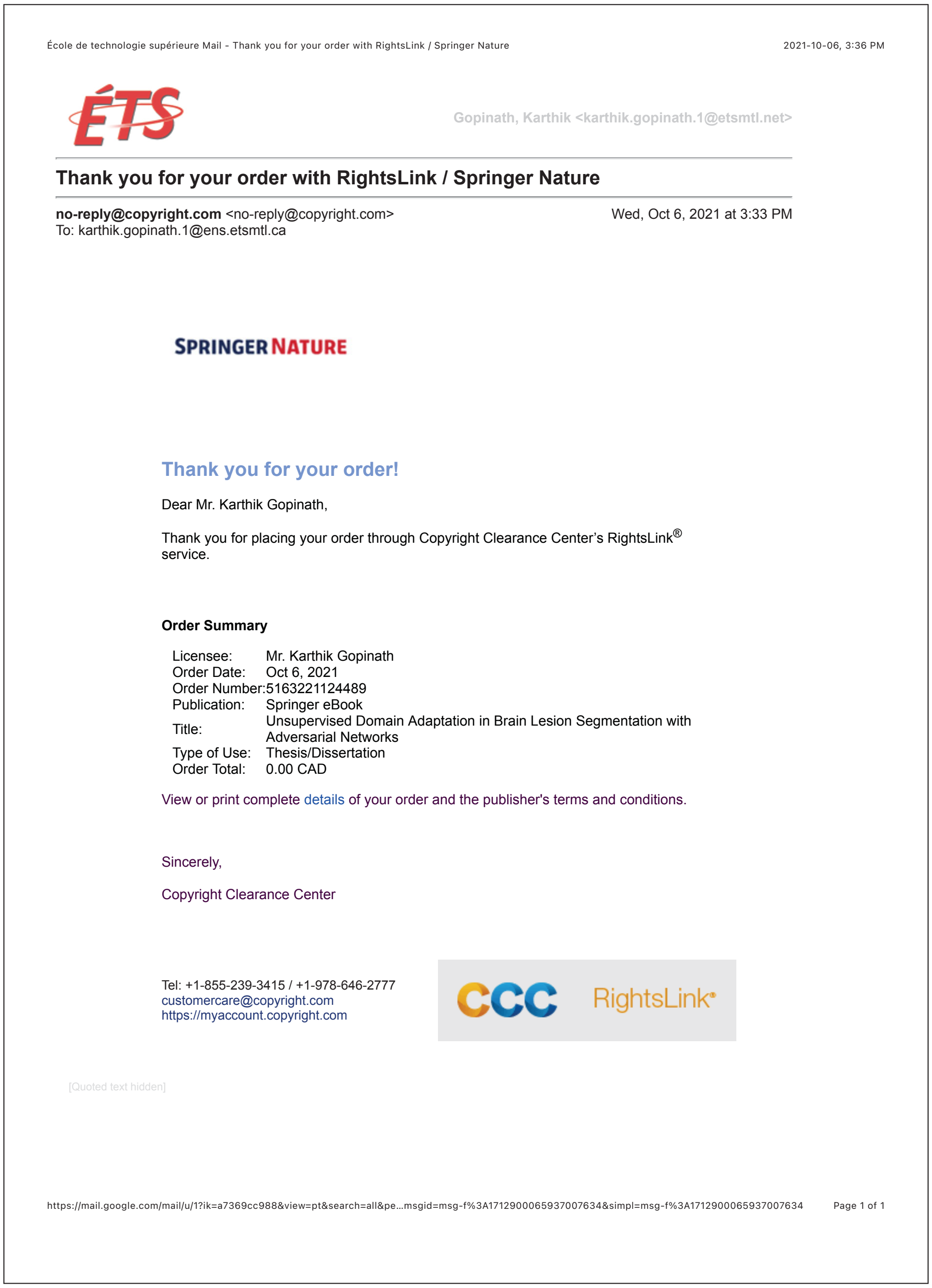
Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

[Quoted text hidden]

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&permmsgid=msg-f%3A1712899912557446541&simpl=msg-f%3A1712899912557446541 Page 1 of 1

Figure-A III-8 Copyright permissions for Figure 1.15



École de technologie supérieure Mail - Thank you for your order with RightsLink / Springer Nature 2021-10-06, 3:36 PM

Gopinath, Karthik <karthik.gopinath.1@etsmtl.net>

Thank you for your order with RightsLink / Springer Nature

no-reply@copyright.com <no-reply@copyright.com> Wed, Oct 6, 2021 at 3:33 PM
To: karthik.gopinath.1@ens.etsmtl.ca

Thank you for your order!

Dear Mr. Karthik Gopinath,

Thank you for placing your order through Copyright Clearance Center's RightsLink® service.

Order Summary

Licensee:	Mr. Karthik Gopinath
Order Date:	Oct 6, 2021
Order Number:	5163221124489
Publication:	Springer eBook
Title:	Unsupervised Domain Adaptation in Brain Lesion Segmentation with Adversarial Networks
Type of Use:	Thesis/Dissertation
Order Total:	0.00 CAD

View or print complete details of your order and the publisher's terms and conditions.

Sincerely,

Copyright Clearance Center

Tel: +1-855-239-3415 / +1-978-646-2777
customercare@copyright.com
https://myaccount.copyright.com

[Quoted text hidden]

https://mail.google.com/mail/u/1?ik=a7369cc988&view=pt&search=all&pe…msgid=msg-f%3A1712900065937007634&simpl=msg-f%3A1712900065937007634 Page 1 of 1

Figure-A III-9 Copyright permissions for Figure 1.17

BIBLIOGRAPHY

Ahmedt-Aristizabal, D., Armin, M. A., Denman, S., Fookes, C. & Petersson, L. (2021). *Graph-Based Deep Learning for Medical Diagnosis and Analysis: Past, Present and Future*.

Anatomy and Physiology. (2013). Available at: https://en.wikipedia.org/wiki/Human_brain, Last accessed on 12/10/2021.

Arbabshirani, M. R., Plis, S., Sui, J. & Calhoun, V. D. (2017). Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls. *NeuroImage*, 145, 137–165.

Arjovsky, M., Chintala, S. & Bottou, L. (2017). Wasserstein generative adversarial networks. *International conference on machine learning*, pp. 214–223.

Atwood, J. & Towsley, D. (2016). Diffusion-convolutional neural networks. *Advances in Neural Information Processing Systems*, pp. 1993–2001.

Auzias, G., Lefèvre, J., Le Troter, A., Fischer, C., Perrot, M., Régis, J. & Coulon, O. (2013). Model-driven harmonic parameterization of the cortical surface: HIP-HOP. *IEEE transactions on medical imaging*, 32(5), 873–887.

Auzias, G., Brun, L., Deruelle, C. & Coulon, O. (2015). Deep sulcal landmarks: Algorithmic and conceptual improvements in the definition and extraction of sulcal pits. *Neuroimage*, 111, 12–25.

Bateson, M., Kervadec, H., Dolz, J., Lombaert, H. & Ayed, I. B. (2019). Constrained domain adaptation for segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 326–334.

Bazin, P.-L. & Pham, D. L. (2007). Topology correction of segmented medical images using a fast marching algorithm. *Computer methods and programs in biomedicine*, 182–190.

Belkin, M., Niyogi, P. & Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR*, 2399–2434.

Bhugra, D. (2005). The global prevalence of schizophrenia. *PLoS medicine*, 2(5), e151.

Boscaini, D., Masci, J., Rodolà, E. & Bronstein, M. (2016). Learning shape correspondence with anisotropic convolutional neural networks. *NIPS*, pp. 3189–3197.

Boykov, Y. & Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on PAMI*, 26(9), 1124–1137.

Bronstein, A. M., Bronstein, M. M. & Kimmel, R. (2006). Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Sciences*, 103(5), 1168–1172.

Bronstein, A. M., Bronstein, M. M. & Kimmel, R. (2007). Calculus of nonrigid surfaces for geometry and texture manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 13(5), 902–913.

Bronstein, M., Bruna, J., LeCun, Y., Szlam, A. & Vandergheynst, P. (2017). Geometric Deep Learning: Going beyond Euclidean data. *IEEE Transactions on Signal Processing*, 34(4), 18–42.

Bronstein, M., Glashoff, K. & Loring, T. (2013). Making Laplacians commute.

Bruce Blaus. (2014). Medical gallery of Blausen Medical 2014. *WikiJournal of Medicine*, 1(2). doi: 10.15347/wjm/2014.010.

Bruna, J., Zaremba, W., Szlam, A. & LeCun, Y. (2014a). Spectral Networks and Locally Connected Networks on Graphs. *ICLR*.

Bruna, J., Zaremba, W., Szlam, A. & Lecun, Y. (2014b). Spectral networks and locally connected networks on graphs. *ICLR*, pp. 1-1.

Budday, S., Steinmann III, P. & Kuhl, E. (2015). Physical biology of human brain development. *Frontiers in cellular neuroscience*, 9, 257.

Buffalo, E. A., Movshon, J. A. & Wurtz, R. H. (2019). From basic brain research to treating human brain disorders. *Proceedings of the National Academy of Sciences*, 116(52), 26167–26172.

Butte, D. D. (2021). LibreTexts Human Anatomy. Available at: https://commons.wikimedia.org/wiki/File:Superior_View_of_the_Brain.jpg, Last accessed on 12/10/2021.

Cacioppo, J. T., Berntson, G. G. & Nusbaum, H. C. (2008). Neuroimaging as a new tool in the toolbox of psychological science. *Current Directions in Psychological Science*, 17(2), 62–67.

Chen, H., Qi, X., Yu, L. & Heng, P.-A. (2016). Dcan: Deep contour-aware networks for accurate gland segmentation. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2487–2496.

Chen, H., Dou, Q., Yu, L., Qin, J. & Heng, P.-A. (2018a). VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. *NeuroImage*, 170, 446–455.

Chen, J., Zhu, J. & Song, L. (2018b). Stochastic Training of Graph Convolutional Networks with Variance Reduction. *International Conference on Machine Learning*, pp. 941–949.

Chen, Y., Li, W. & Van Gool, L. (2018c). Road: Reality oriented adaptation for semantic segmentation of urban scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7892–7901.

Chu, S.-H., Parhi, K. K. & Lenglet, C. (2018). Function-specific and enhanced brain structural connectivity mapping via joint modeling of diffusion and functional MRI. *Scientific reports*, 8(1), 1–19.

Chung, F. (1997). *Spectral Graph Theory*. AMS.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T. & Ronneberger, O. (2016). 3D U-Net: learning dense volumetric segmentation from sparse annotation. *MICCAI*, pp. 424–432.

Ciresan, D., Giusti, A., Gambardella, L. M. & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in neural information processing systems*, pp. 2843–2851.

Cointepas, Y., Geffroy, D., Souedet, N., Denghien, I. & Rivière, D. (2010). The BrainVISA project: a shared software development infrastructure for biomedical imaging research. *HBM*, 16, -.

Cornell, B. (2016). Brain Matter. Available at: https://ib.bioninja.com.au/options/option-a-neurobiology-and/a2-the-human-brain/brain-matter.html, Last accessed on 12/10/2021.

Craddock, R. C., James, G. A., Holtzheimer III, P. E., Hu, X. P. & Mayberg, H. S. (2012). A whole brain fMRI atlas generated via spatially constrained spectral clustering. *Human brain mapping*, 33(8), 1914–1928.

Cruz, R. S. (2021). Slides – DeepCSR: A 3D deep learning approach for cortical surface reconstruction. [Online; accessed 15-October-2021], Consulted at https://rfsantacruz.com/files/pdfs/wacv21_deepcsr_slides.pdf.

Cruz, R. S., Lebrat, L., Bourgeat, P., Fookes, C., Fripp, J. & Salvado, O. (2021). DeepCSR: A 3D deep learning approach for cortical surface reconstruction. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 806–815.

Cucurull, G., Wagstyl, K., Casanova, A., Veličković, P., Jakobsen, E., Drozdzal, M., Romero, A., Evans, A. & Bengio, Y. (2018). Convolutional neural networks for mesh-based parcellation of the cerebral cortex. *MIDL*, pp. 1.

Dahnke, R., Yotter, R. A. & Gaser, C. (2013). Cortical thickness and central surface estimation. *Neuroimage*, 336–348.

Dale, A. M., Fischl, B. & Sereno, M. I. (1999). Cortical surface-based analysis: I. Segmentation and surface reconstruction. *Neuroimage*, 179–194.

Defferrard, M., Bresson, X. & Vandergheynst, P. (2016). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *NIPS*, pp. 3844–3852.

Delisle, P.-L., Anctil-Robitaille, B., Desrosiers, C. & Lombaert, H. (2021). Realistic image normalization for multi-Domain segmentation. *Medical Image Analysis*, 74, 102191. doi: https://doi.org/10.1016/j.media.2021.102191.

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. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *Neuroimage*, 31(3), 968–980.

Destrieux, C., Fischl, B., Dale, A. & Halgren, E. (2009). A sulcal depth-based anatomical parcellation of the cerebral cortex. *NeuroImage*, 47, 151–163.

DeYoe, E. A., Bandettini, P., Neitz, J., Miller, D. & Winans, P. (1994). Functional magnetic resonance imaging of the human brain. *Journal of neuroscience methods*, 54(2), 171–187.

Dgani, Y., Greenspan, H. & Goldberger, J. (2018). Training a neural network based on unreliable human annotation of medical images. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 39–42.

Dhillon, I. S., Guan, Y. & Kulis, B. (2007). Weighted Graph Cuts without Eigenvectors A Multilevel Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11), 1944-1957. doi: 10.1109/TPAMI.2007.1115.

Di Martino, A., Yan, C.-G., Li, Q., Denio, E., Castellanos, F. X., Alaerts, K., Anderson, J. S., Assaf, M., Bookheimer, S. Y., Dapretto, M. et al. (2014). The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6), 659–667.

DiLuca, M. & Olesen, J. (2014). The cost of brain diseases: a burden or a challenge? *Neuron*, 82(6), 1205–1208.

Dolz, J., Desrosiers, C. & Ben Ayed, I. (2017). 3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study. *NeuroImage*, 170, 456–470.

Dong, J., Cong, Y., Sun, G., Yang, Y., Xu, X. & Ding, Z. (2020). Weakly-supervised cross-domain adaptation for endoscopic lesions segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(5), 2020-2033.

Dou, Q., Chen, H., Jin, Y., Yu, L., Qin, J. & Heng, P.-A. (2016). 3d deeply supervised network for automatic liver segmentation from ct volumes. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 149–157.

Dou, Q., Ouyang, C., Chen, C., Chen, H. & Heng, P.-A. (2018). Unsupervised cross-modality domain adaptation of convnets for biomedical image segmentations with adversarial loss. *IJCAI International Joint Conference on Artificial Intelligence*, pp. 691.

Epstein, J., Stern, E. & Silbersweig, D. (2001). Neuropsychiatry at the millenium: the potential for mind/brain integration through emerging interdisciplinary research strategies. *Clinical Neuroscience Research*, 1(1-2), 10–18.

Ernst, R. L. & Hay, J. W. (1994). The US economic and social costs of Alzheimer's disease revisited. *American Journal of Public Health*, 84(8), 1261–1264.

Eynard, D., Kovnatsky, A., Bronstein, M. M., Glashoff, K. & Bronstein, A. M. (2015). Multimodal Manifold Analysis by Simultaneous Diagonalization of Laplacians. *IEEE Transactions on PAMI*, 12, 2505–2517.

Fan, J., Cao, X., Xue, Z., Yap, P.-T. & Shen, D. (2018). Adversarial Similarity Network for Evaluating Image Alignment in Deep Learning based Registration. 11070, 739–746.

Fejtek, M. (2019). Available at: https://www.neuroanatomy.ca/regions/cortex/cortex_lobes.png, Last accessed on 12/10/2021.

Fey, M., Lenssen, J. E., Weichert, F. & Müller, H. (2018a). SplineCNN: Fast Geometric Deep Learning with Continuous B-Spline Kernels. *CVPR*.

Fey, M., Lenssen, J. E., Weichert, F. & Müller, H. (2018b). SplineCNN: Fast Geometric Deep Learning with Continuous B-Spline Kernels. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Fischl, B., van der Kouwe, A., Destrieux, C., Halgren, E., Segonne, F., Salat, D. H., Busa, E., Seidman, L. J., Goldstein, J., Kennedy, D., Caviness, V., Makris, N., Rosen, B. & Dale, A. M. (2004a). Automatically parcellating the human cerebral cortex. *Cerebral cortex*, 14(1), 11–22.

Fischl, B. et al. (2004b). Automatically parcellating the cortex. *Cereb. Cortex*, 11–22.

Fischl, B. & Dale, A. M. (2000). Measuring the thickness of the human cerebral cortex from magnetic resonance images. *Proceedings of the National Academy of Sciences*, 97(20), 11050–11055.

Fischl, B., Sereno, M. I. & Dale, A. M. (1999). Cortical surface-based analysis: II: inflation, flattening, and a surface-based coordinate system. *Neuroimage*, 195–207.

Freeborough, P. A. & Fox, N. C. (1998). MR image texture analysis applied to the diagnosis and tracking of Alzheimer's disease. *TMI*, 475–478.

Ganin, Y. & Lempitsky, V. (2015). Unsupervised Domain Adaptation by Backpropagation. *International Conference on Machine Learning*, pp. 1180–1189.

Gao, H. & Ji, S. (2019). Graph U-Nets. *ICML*, pp. 2083–2092.

Geffroy, D., Rivière, D., Denghien, I., Souedet, N., Laguitton, S. & Cointepas, Y. (2011). BrainVISA: a complete software platform for neuroimaging. *Python in Neuroscience workshop, Paris*.

Germanaud, D., Lefèvre, J., Toro, R., Fischer, C., Dubois, J., Hertz-Pannier, L. & Mangin, J. F. (2012). Larger is twistier: Spectral analysis of gyrification (SPANGY) applied to adult brain size polymorphism. *NeuroImage*, 63, 1257–1272.

Ghafoorian, M., Mehrtash, A., Kapur, T., Karssemeijer, N., Marchiori, E., Pesteie, M., Guttmann, C. R., de Leeuw, F.-E., Tempany, C. M., van Ginneken, B. et al. (2017). Transfer learning for domain adaptation in mri: Application in brain lesion segmentation. *International conference on medical image computing and computer-assisted intervention*, pp. 516–524.

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. (2016). A multi-modal parcellation of human cerebral cortex. *Nature*, 536, 171–178.

Glaunes, J., Trouve, A. & Younes, L. (2004). Diffeomorphic matching of distributions: a new approach for unlabelled point-sets and sub-manifolds matching. *CVPR*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, pp. 2672–2680.

Gopinath, K., Desrosiers, C. & Lombaert, H. (2018). Graph Convolutions on Spectral Embeddings: Learning of Cortical Surface Data. *arXiv preprint 1803.10336*.

Gopinath, K., Desrosiers, C. & Lombaert, H. (2019a). Adaptive Graph Convolution Pooling for Brain Surface Analysis. *International Conference on Information Processing in Medical Imaging*, pp. 86–98.

Gopinath, K., Desrosiers, C. & Lombaert, H. (2019b). Graph convolutions on spectral embeddings for cortical surface parcellation. *MedIA*, 297–305.

Gopinath, K., Desrosiers, C. & Lombaert, H. (2020a). Graph Domain Adaptation for Alignment-Invariant Brain Surface Segmentation. In *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Graphs in Biomedical Image Analysis* (pp. 152–163). Springer.

Gopinath, K., Desrosiers, C. & Lombaert, H. (2020b). Learnable Pooling in Graph Convolution Networks for Brain Surface Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1-1. doi: 10.1109/TPAMI.2020.3028391.

Grady, L. & Polimeni, J. R. (2010). *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer.

Gray, H. (1918). Anatomy of the Human Body. Available at: https://commons.wikimedia.org/wiki/File:Gray677.png, Last accessed on 12/10/2021.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L. & Bennamoun, M. (2020). Deep Learning for 3D Point Clouds: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1-1. doi: 10.1109/TPAMI.2020.3005434.

Hamilton, W. L., Ying, R. & Leskovec, J. (2017). Inductive Representation Learning on Large Graphs. *NIPS*.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M. & Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35, 18–31.

He, R., Gopinath, K., Desrosiers, C. & Lombaert, H. (2020). Spectral Graph Transformer Networks for Brain Surface Parcellation. *ISBI*, pp. 372–376.

Heinrichs, R. W. & Zakzanis, K. K. (1998). Neurocognitive deficit in schizophrenia: a quantitative review of the evidence. *Neuropsychology*, 12(3), 426.

Henaff, M., Bruna, J. & LeCun, Y. (2015). Deep Convolutional Networks on Graph-Structured Data. abs/1506.05163, 1-1. Consulted at http://arxiv.org/abs/1506.05163.

Henschel, L., Conjeti, S., Estrada, S., Diers, K., Fischl, B. & Reuter, M. (2020). FastSurfer-A fast and accurate deep learning based neuroimaging pipeline. *NeuroImage*, 117012.

Hoffman, J., Wang, D., Yu, F. & Darrell, T. (2016). Fcns in the wild: Pixel-level adversarial and constraint-based adaptation.

Hong, W., Wang, Z., Yang, M. & Yuan, J. (2018). Conditional generative adversarial network for structured domain adaptation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1335–1344.

Hua, X., Hibar, D. P., Ching, C. R., Boyle, C. P., Rajagopalan, P., Gutman, B. A., Leow, A. D., Toga, A. W., Jack, C. R., Harvey, D., Weiner, M. W., Thompson, P. M. & Alzheimer's Disease Neuroimaging Initiative. (2013). Unbiased tensor-based morphometry: Improved robustness and sample size estimates for Alzheimer's disease clinical trials. *NeuroImage*, 66, 648–661.

Huang, G., Liu, Z., van der Maaten, L. & Weinberger, K. Q. (2017). Densely Connected Convolutional Networks. *CVPR*, 1, 1-10.

Islam, J. & Zhang, Y. (2018). Brain MRI analysis for Alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks. *Brain informatics*, 5(2), 1–14.

Jack, C. R. et al. (2008). ADNI: MRI methods. *JMRI*, 685–691.

Jaderberg, M., Simonyan, K., Zisserman, A. et al. (2015). Spatial transformer networks. *NIPS*, pp. 2017–2025.

Javan, R., Schickel, M., Zhao, Y., Agbo, T., Fleming, C., Heidari, P., Gholipour, T., Shields, D. C. & Koubeissi, M. (2020). Using 3D-printed mesh-like brain cortex with deep structures for planning intracranial EEG electrode placement. *Journal of digital imaging*, 33(2), 324–333.

Javanmardi, M. & Tasdizen, T. (2018). Domain adaptation for biomedical image segmentation using adversarial training. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 554–558.

Kamnitsas, K., Bai, W., Ferrante, E., McDonagh, S., Sinclair, M., Pawlowski, N., Rajchl, M., Lee, M., Kainz, B., Rueckert, D. & Glocker, B. (2018). Ensembles of multiple models and architectures for robust brain tumour segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10670 LNCS, 450–462.

Kamnitsas, K., Baumgartner, C., Ledig, C., Newcombe, V., Simpson, J., Kane, A., Menon, D., Nori, A., Criminisi, A., Rueckert, D. et al. (2017a). Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. *International conference on information processing in medical imaging*, pp. 597–609.

Kamnitsas, K., Ledig, C., Newcombe, V. F. J., Simpson, J. P., Kane, A. D., Menon, D. K., Rueckert, D. & Glocker, B. (2017b). Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Medical Image Analysis*, 36, 61–78.

Kazi, A., Shekarforoush, S., Krishna, S. A., Burwinkel, H., Vivar, G., Kortüm, K., Ahmadi, S.-A., Albarqouni, S. & Navab, N. (2019). Inceptiongcn: receptive field aware graph convolutional network for disease prediction. *International Conference on Information Processing in Medical Imaging*, pp. 73–85.

Kazi, A., Cosmo, L., Navab, N. & Bronstein, M. (2020). *Differentiable graph module (DGM) for graph convolutional networks*.

Kessler, R. C., McGonagle, K. A., Zhao, S., Nelson, C. B., Hughes, M., Eshleman, S., Wittchen, H.-U. & Kendler, K. S. (1994). Lifetime and 12-month prevalence of DSM-III-R psychiatric disorders in the United States: results from the National Comorbidity Survey. *Archives of general psychiatry*, 51(1), 8–19.

Kessler, R. C., Berglund, P., Demler, O., Jin, R., Koretz, D., Merikangas, K. R., Rush, A. J., Walters, E. E. & Wang, P. S. (2003). The epidemiology of major depressive disorder: results from the National Comorbidity Survey Replication (NCS-R). *Jama*, 289(23), 3095–3105.

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. (2005). Automated 3-D extraction and evaluation of the inner and outer cortical surfaces using a Laplacian map and partial volume effect classification. *Neuroimage*, 210–221.

Kingma, D. P. & Ba, J. (2014). Adam: Stochastic optimization. *ICLR*.

Kipf, T. N. (2016). Github Blog – Graph Convolutional Network. [Online; accessed 15-October-2021], Consulted at http://tkipf.github.io/graph-convolutional-networks/.

Kipf, T. N. & Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. *ICLR*.

Klein, A. & Tourville, J. (2012). 101 labeled brain images and a consistent human cortical labeling protocol. *Frontiers in neuroscience*, 171.

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. (2017). Mindboggling morphometry of human brains. *PLOS Computational Biology*, 13(2), e1005350.

Kokkinos, I., Bronstein, M. M., Litman, R. & Bronstein, A. M. (2012). Intrinsic shape context descriptors for deformable shapes. *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 159–166.

Kovnatsky, A., Bronstein, M. M., Bronstein, A. M., Glashoff, K. & Kimmel, R. (2013). Coupled quasi-harmonic bases. *Computer Graphics Forum*, 32, 439–448.

Kriegeskorte, N. & Goebel, R. (2001). An efficient algorithm for topologically correct segmentation of the cortical sheet in anatomical MR volumes. *NeuroImage*, 329–346.

Ktena, S. I., Parisot, S., Ferrante, E., Rajchl, M., Lee, M., Glocker, B. & Rueckert, D. (2017). Distance Metric Learning using Graph Convolutional Networks: Application to Functional Brain Networks. *MICCAI*, 10433, 469–477.

Ktena, S. I., Parisot, S., Ferrante, E., Rajchl, M., Lee, M., Glocker, B. & Rueckert, D. (2018). Metric learning with spectral graph convolutions on brain connectivity networks. *NeuroImage*, 169, 431–442.

Le Bihan, D., Mangin, J.-F., Poupon, C., Clark, C. A., Pappata, S., Molko, N. & Chabriat, H. (2001). Diffusion tensor imaging: concepts and applications. *Journal of magnetic resonance imaging*, 13(4), 534–546.

Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition. *IEEE Intelligent Signal Processing*, 86(11), 2278–2324.

Ledig, C. et al. (2014). Alzheimer's state classification using volumetry, thickness and intensity. *MICCAI*.

Lefèvre, J., Pepe, A., Muscato, J., De Guio, F., Girard, N., Auzias, G. & Germanaud, D. (2018). SPANOL (SPectral ANalysis of Lobes): A Spectral Clustering Framework for Individual and Group Parcellation of Cortical Surfaces in Lobes. *Frontiers in Neuroscience*, 12, 354–366.

Lerch, J. P. et al. (2004). Focal decline of cortical thickness in Alzheimer's disease identified by computational neuroanatomy. *Cereb. cortex*, 995–1001.

Levie, R., Monti, F., Bresson, X. & Bronstein, M. M. (2018). CayleyNets: Graph Convolutional Neural Networks with Complex Rational Spectral Filters. *ICLR*.

Lewinsohn, P. M., Duncan, E. M., Stanton, A. K. & Hautzinger, M. (1986). Age at first onset for nonbipolar depression. *Journal of Abnormal Psychology*, 95(4), 378.

Li, G., Wang, L., Shi, F., Gilmore, J. H., Lin, W. & Shen, D. (2015). Construction of 4D high-definition cortical surface atlases of infants: Methods and applications. *Medical Image Analysis*, 25(1), 22–36.

Li, M., Lian, X.-C., Kwok, J. T. & Lu, B.-L. (2011). Time and space efficient spectral clustering via column sampling. *CVPR 2011*, pp. 2297–2304.

Liang, Z.-P. & Lauterbur, P. C. (2000). *Principles of magnetic resonance imaging: a signal processing perspective*. SPIE Optical Engineering Press.

Lohmann, G., Von Cramon, D. Y. & Colchester, A. C. (2007). Deep sulcal landmarks provide an organizing framework for human cortical folding. *Cerebral Cortex*, 18(6), 1415–1420.

Lombaert, H., Grady, L., Polimeni, J. R. & Cheriet, F. (2011). Fast brain matching with spectral correspondence. *Biennial International Conference on Information Processing in Medical Imaging*, pp. 660–673.

Lombaert, H., Grady, L., Polimeni, J. R. & Cheriet, F. (2012). FOCUSR: feature oriented correspondence using spectral regularization–a method for precise surface matching. *IEEE transactions on pattern analysis and machine intelligence*, 35(9), 2143–2160.

Lombaert, H., Sporring, J. & Siddiqi, K. (2013). Diffeomorphic spectral matching of cortical surfaces. *International Conference on Information Processing in Medical Imaging*, pp. 376–389.

Lombaert, H., Grady, L., Pennec, X., Ayache, N. & Cheriet, F. (2014). Spectral log-demons: diffeomorphic image registration with very large deformations. *International journal of computer vision*, 107(3), 254–271.

Lombaert, H., Arcaro, M. & Ayache, N. (2015a). Brain Transfer: Spectral Analysis of Cortical Surfaces and Functional Maps. *IPMI*, 24, 474–487.

Lombaert, H., Criminisi, A. & Ayache, N. (2015b). Spectral Forests: Learning of Surface Data, Application to Cortical Parcellation. *MICCAI*, 9349, 547–555.

Long, M., Cao, Y., Wang, J. & Jordan, M. (2015). Learning Transferable Features with Deep Adaptation Networks. *International Conference on Machine Learning*, pp. 97–105.

López-López, N., Vázquez, A., Poupon, C., Mangin, J.-F., Ladra, S. & Guevara, P. (2020). GeoSP: A parallel method for a cortical surface parcellation based on geodesic distance. *EMBC*, pp. 1696–1700.

Lorensen, W. E. & Cline, H. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *ACM siggraph computer graphics*, 163–169.

Madani, A., Moradi, M., Karargyris, A. & Syeda-Mahmood, T. (2018). Semi-supervised learning with generative adversarial networks for chest X-ray classification with ability of data domain adaptation. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 1038-1042.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z. & Paul Smolley, S. (2017). Least squares generative adversarial networks. *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802.

Marcus, D. S., Wang, T. H., Parker, J., Csernansky, J. G., Morris, J. C. & Buckner, R. L. (2007). Open Access Series of Imaging Studies (OASIS): Cross-sectional MRI Data in Young, Middle Aged, Nondemented, and Demented Older Adults. *Journal of Cognitive Neuroscience*, 19(9), 1498-1507. doi: 10.1162/jocn.2007.19.9.1498.

Masci, J., Boscaini, D., Bronstein, M. & Vandergheynst, P. (2015). Geodesic convolutional neural networks on Riemannian manifolds. *ICCV-3dRR*, pp. 37–45.

Mateus, D., Horaud, R., Knossow, D., Cuzzolin, F. & Boyer, E. (2008). Articulated shape matching using laplacian eigenfunctions and unsupervised point registration. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.

Merboldt, K.-D., Hanicke, W. & Frahm, J. (1985). Self-diffusion NMR imaging using stimulated echoes. *Journal of Magnetic Resonance (1969)*, 64(3), 479–486.

Milletari, F., Navab, N. & Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. *3D Vision (3DV), 2016 Fourth International Conference on*, pp. 565–571.

Molnar, C., Gair, J. et al. (2015). *Concepts of Biology: 1st Canadian Edition*.

Monti, F., Boscaini, D., Masci, J., Rodolà, E., Svoboda, J. & Bronstein, M. M. (2017). Geometric deep learning on graphs and manifolds using mixture model CNNs. *CVPR*, 1, 1–10.

Murphy, D. G., DeCarli, C., Mclntosh, A. R., Daly, E., Mentis, M. J., Pietrini, P., Szczepanik, J., Schapiro, M. B., Grady, C. L., Horwitz, B. et al. (1996). Sex differences in human brain morphometry and metabolism: an in vivo quantitative magnetic resonance imaging and positron emission tomography study on the effect of aging. *Archives of general psychiatry*, 585–594.

Nair, V. & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *ICML*, pp. 807–814.

Niethammer, M., Reuter, M., Wolter, F.-E., Bouix, S., Peinecke, N., Koo, M.-S. & Shenton, M. E. (2007). Global medical shape analysis using the Laplace-Beltrami spectrum. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 850–857.

Nieuwenhuys, R. (2013). The myeloarchitectonic studies on the human cerebral cortex of the Vogt-Vogt school, and their significance for the interpretation of functional neuroimaging data. In *Microstructural parcellation of the human cerebral cortex* (pp. 55–125). Springer.

Nolte, J. (1993). *The human brain*. Mosby/Elsevier,.

Ogawa, S., Lee, T.-M., Kay, A. R. & Tank, D. W. (1990). Brain magnetic resonance imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences*, 87(24), 9868–9872.

Oliveira Jr, P. P. d. M., Nitrini, R., Busatto, G., Buchpiguel, C., Sato, J. R. & Amaro Jr, E. (2010). Use of SVM methods with surface-based cortical and volumetric subcortical measurements to detect Alzheimer's disease. *Journal of Alzheimer's Disease*, 1263–1272.

Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A. & Guibas, L. (2012). Functional maps: A flexible representation of maps between shapes. *SIGGRAPH*, 31(4), 30.

Parisot, S., Ktena, S. I., Ferrante, E., Lee, M., Moreno, R. G., Glocker, B. & Rueckert, D. (2017). Spectral Graph Convolutions for Population-based Disease Prediction. *MICCAI*, 10435, 177–185.

Parisot, S., Ktena, S. I., Ferrante, E., Lee, M., Guerrero, R., Glocker, B. & Rueckert, D. (2018). Disease Prediction using Graph Convolutional Networks: Application to Autism Spectrum Disorder and Alzheimer's Disease. *Medical Image Analysis*, 48(1), 117–130.

Park, J. J., Florence, P., Straub, J., Newcombe, R. & Lovegrove, S. (2019). Deepsdf: Learning continuous signed distance functions for shape representation. *CVPR*, pp. 165–174.

Perona, P. & Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7), 629–639.

Qi, C. R., Su, H., Mo, K. & Guibas, L. J. (2017a). Pointnet: Deep learning on point sets for 3d classification and segmentation. *CVPR*, pp. 652–660.

Qi, C. R., Yi, L., Su, H. & Guibas, L. J. (2017b). PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *Advances in Neural Information Processing Systems*, 30, 5099–5108.

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. (2009). Early diagnosis of Alzheimer's disease using cortical thickness: impact of cognitive reserve. *Brain*, 2036–2047.

Rakhimberdina, Z. & Murata, T. (2019). Linear graph convolutional model for diagnosing brain disorders. *International Conference on Complex Networks and Their Applications*, pp. 815–826.

Razavi, M. J., Zhang, T., Liu, T. & Wang, X. (2015). Cortical folding pattern and its consistency induced by biological growth. *Scientific reports*, 5(1), 1–14.

Reuter, M. (2010). Hierarchical shape segmentation and registration via topological features of Laplace-Beltrami eigenfunctions. *International Journal of Computer Vision*, 89(2-3), 287–308.

Reuter, M., Wolter, F.-E. & Peinecke, N. (2006). Laplace–Beltrami spectra as 'Shape-DNA'of surfaces and solids. *Computer-Aided Design*, 38(4), 342–366.

Rice, D. P. (1999). The economic impact of schizophrenia. *Journal of Clinical Psychiatry*, 60(1), 4–6.

Rivière, D., Régis, J., Cointepas, Y., Papadopoulos-Orfanos, D., Cachia, A. & Mangin, J. F. (2003). A freely available Anatomist-BrainVISA package for structural morphometry of the cortical sulci. *Neuroimage*, 19(2), 934.

Ronneberger, O., Fischer, P. & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, 9351, 234–241.

Saladin, K. S. & Miller, L. (1998). *Anatomy & physiology*. WCB/McGraw-Hill New York.

Sánchez, I. & Vilaplana, V. (2018). Brain MRI super-resolution using 3D generative adversarial networks. *arXiv preprint arXiv:1812.11440*, 1–8.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M. & Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1), 61–80.

Ségonne, F., Dale, A. M., Busa, E., Glessner, M., Salat, D., Hahn, H. K. & Fischl, B. (2004). A hybrid approach to the skull stripping problem in MRI. *Neuroimage*, 22(3), 1060–1075.

Selvan, R., Kipf, T., Welling, M., Juarez, A. G.-U., Pedersen, J. H., Petersen, J. & de Bruijne, M. (2020). Graph refinement based airway extraction using mean-field networks and graph neural networks. *Medical Image Analysis*, 64, 101751.

Shapiro, L. S. & Brady, J. M. (1992). Feature-based correspondence: an eigenvector approach. 10(5), 283–288.

Sharma, A., Horaud, R. & Mateus, D. (2021). 3D shape registration using spectral graph embedding and probabilistic matching. pp. 1–1.

Shattuck, D. W. & Leahy, R. M. (2002). BrainSuite: an automated cortical surface identification tool. *Medical image analysis*, 129–142.

Shen, D., Wu, G. & Suk, H.-I. (2017). Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19, 221–248.

Shipp, S. (2007). Structure and function of the cerebral cortex. *Current Biology*, 17(12), R443–R449.

Shuman, D. I., Vandergheynst, P., Kressner, D. & Frossard, P. (2011). Distributed Signal Processing via Chebyshev Polynomial Approximation. *Conference on Distributed Computing in Sensor Systems (DCOSS)*.

Simonovsky, M. & Komodakis, N. (2017). Dynamic Edge-Conditioned Filters in Convolutional Neural Networks on Graphs. *CVPR*, 1-1, 29–38.

Song, T., Zheng, W., Song, P. & Cui, Z. (2018). EEG emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3), 532–541.

Sowell, E. R. et al. (2004). Longitudinal mapping of cortical thickness and brain growth in normal children. *Journal of Neuroscience*, 8223–8231.

Spielman, D. (2012). Spectral graph theory. *Combinatorial scientific computing*, 18, 1-1.

Styner, M., Oguz, I., Xu, S., Brechbühler, C., Pantazis, D., Levitt, J. J., Shenton, M. E. & Gerig, G. (2006). Framework for the Statistical Shape Analysis of Brain Structures using SPHARM-PDM. *The insight journal*, (1071), 242.

Sun, J., Ovsjanikov, M. & Guibas, L. (2009). A concise and provably informative multi-scale signature based on heat diffusion. *Computer graphics forum*, 28(5), 1383–1392.

Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J. N., Wu, Z. & Ding, X. (2020). Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. *Medical Image Analysis*, 63, 101693.

Tallinen, T., Chung, J. Y., Rousseau, F., Girard, N., Lefèvre, J. & Mahadevan, L. (2016). On the growth and form of cortical convolutions. *Nature Physics*, 12(6), 588–593.

Tang, X., Holland, D., Dale, A. M., Younes, L., Miller, M. I. & Initiative, A. D. N. (2014). Shape abnormalities of subcortical and ventricular structures in mild cognitive impairment and Alzheimer's disease: detecting, quantifying, and predicting. *Human brain mapping*, 3701–3725.

Tosun, D., Rettmann, M. E., Naiman, D. Q., Resnick, S. M., Kraut, M. A. & Prince, J. L. (2006). Cortical reconstruction using implicit surface evolution: Accuracy and precision analysis. *NeuroImage*, 29(3), 838-852. doi: https://doi.org/10.1016/j.neuroimage.2005.08.061.

Tsai, Y.-H., Hung, W.-C., Schulter, S., Sohn, K., Yang, M.-H. & Chandraker, M. (2018). Learning to adapt structured output space for semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7472–7481.

Tustison, N. J., Cook, P. A., Klein, A., Song, G., Das, S. R., Duda, J. T., Kandel, B. M., van Strien, N., Stone, J. R., Gee, J. C. & Avants, B. B. (2014). Large-scale evaluation of ANTs and FreeSurfer cortical thickness measurements. *NeuroImage*, 99, 166–179.

Tymofiyeva, O. & Gaschler, R. (2021). Training-Induced Neural Plasticity in Youth: A Systematic Review of Structural and Functional MRI Studies. *Frontiers in Human Neuroscience*, 14, 579. doi: 10.3389/fnhum.2020.497245.

Valverde, S., Cabezas, M., Roura, E., González-Villà, S., Pareto, D., Vilanova, J. C., Ramió-Torrentà, L., Rovira, À., Oliver, A. & Lladó, X. (2017). Improving automated multiple sclerosis lesion segmentation with a cascaded 3D convolutional neural network approach. *NeuroImage*, 155, 159–168.

Van Essen, D. C., Glasser, M. F., Dierker, D. L., Harwell, J. & Coalson, T. (2011). Parcellations and hemispheric asymmetries of human cerebral cortex analyzed on surface-based atlases. *Cerebral cortex*, 22(10), 2241–2262.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P. & Bengio, Y. (2018). Graph Attention Networks. *ICLR*, pp. 1.

Vemuri, P., Gunter, J. L., Senjem, M. L., Whitwell, J. L., Kantarci, K., Knopman, D. S., Boeve, B. F., Petersen, R. C. & Jack Jr, C. R. (2008). Alzheimer's disease diagnosis in individual subjects using structural MR images: validation studies. *Neuroimage*, 1186–1197.

Vu, T.-H., Jain, H., Bucher, M., Cord, M. & Pérez, P. (2019). Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2517–2526.

Wachinger, C., Golland, P., Kremen, W., Fischl, B., Reuter, M., Initiative, A. D. N. et al. (2015). BrainPrint: A discriminative characterization of brain morphology. *NeuroImage*, 109, 232–248.

Wachinger, C., Reuter, M. & Klein, T. (2017). DeepNAT: Deep convolutional neural network for segmenting neuroanatomy. *NeuroImage*, 170, 434-445.

Wang, C., Samari, B. & Siddiqi, K. (2018). Local Spectral Graph Convolution for Point Set Feature Learning. *ECCV*.

Wikimedia Commons. (2011). Available at: https://commons.wikimedia.org/wiki/File:Human_motor_cortex_topography.png, Last accessed on 12/10/2021.

Witten, I. H., Frank, E., Hall, M. A. & Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wu, Z., Zhao, F., Xia, J., Wang, L., Lin, W., Gilmore, J. H., Li, G. & Shen, D. (2019). Intrinsic Patch-Based Cortical Anatomical Parcellation Using Graph Convolutional Neural Network on Surface Manifold. *MICCAI*, pp. 492–500.

Xu, Y., Fan, T., Xu, M., Zeng, L. & Qia, Y. (2018). SpiderCNN: Deep Learning on Point Sets with Parameterized Convolutional Filters. *ECCV*.

Yan, Z., Youyong, K., Jiasong, W., Coatrieux, G. & Huazhong, S. (2019). Brain tissue segmentation based on graph convolutional networks. *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 1470–1474.

Yang, D., Xu, D., Zhou, S. K., Georgescu, B., Chen, M., Grbic, S., Metaxas, D. & Comaniciu, D. (2017). Automatic liver segmentation using an adversarial image-to-image network. *Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2017*, pp. 507–515.

Yeo, B. T., Sabuncu, M. R., Vercauteren, T., Ayache, N., Fischl, B. & Golland, P. (2010). Spherical demons: Fast diffeomorphic landmark-free surface registration. *IEEE Transactions on Medical Imaging*, 29(3), 650–668.

Yi, L., Su, H., Guo, X. & Guibas, L. J. (2017). SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation. *CVPR*.

Yi, X., Walia, E. & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, 58, 101552.

Ying, R., You, J., Morris, C., Ren, X., Hamilton, W. L. & Leskovec, J. (2018). Hierarchical Graph Representation Learning with Differentiable Pooling. *NeurIPS*.

Zhang, T. & Davatzikos, C. (2011). ODVBA: Optimally-discriminative voxel-based analysis. *IEEE Transactions on Medical Imaging*, 30(8), 1441–1454.

Zhang, Y., David, P. & Gong, B. (2017). Curriculum domain adaptation for semantic segmentation of urban scenes. *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2020–2030.

Zhang, Y., Miao, S., Mansi, T. & Liao, R. (2018). Task driven generative modeling for unsupervised domain adaptation: Application to x-ray image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 599–607.

Zhao, F., Wu, Z., Wang, L., Lin, W., Gilmore, J. H., Xia, S., Shen, D. & Li, G. (2021). Spherical Deformable U-Net: Application to Cortical Surface Parcellation and Development Prediction. *IEEE transactions on medical imaging*, 40(4), 1217–1228.

Zhao, X., Chen, F., Hu, S. & Cho, J.-H. (2020). Uncertainty Aware Semi-Supervised Learning on Graph Data. *Advances in Neural Information Processing Systems*, 33, 1-1.

Zou, Y., Yu, Z., Vijaya Kumar, B. & Wang, J. (2018). Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. *Proceedings of the European conference on computer vision (ECCV)*, pp. 289–305.