UNIVERSITY OF COPENHAGEN
FACULTY OF SCIENCE

PhD thesis

# Group and Pseudo-group Convolutional Neural Networks

## Learning on Curved Spaces with the Application to DWI Segmentation

Renfei Liu

**Advisors: Sune Darkner, François Lauze, Kenny Erleben**

**Submitted: March 11, 2022**

This thesis has been submitted to the PhD School of The Faculty of Science, University of Copenhagen

# Contents

**Contents** **0**
Abstract . . . . . i
Dansk Resumé . . . . . iii
Acknowledgements . . . . . v

**1 Introduction** **1**
1.1 Motivation and Problem Definition . . . . . 1
1.1.1 The underlying space may not have a group structure. . 1
1.1.2 Motions that come with the data. . . . . 2
1.1.3 Problem statement . . . . . 3
1.2 Data Background: Diffusion Weighted Imaging (DWI) . . . . . 3
1.2.1 DWI Basics . . . . . 3
1.2.2 Mathematical Interpretation of DWI . . . . . 4

**2 CNN Background: Geometric Deep Learning and Equivariant Neural Networks** **7**
2.1 Generalizing Convolution to Group Actions . . . . . 7
2.1.1 $G$-action, Homogeneous spaces . . . . . 8
2.1.2 Equivariant Map . . . . . 8
2.1.3 Quotient space . . . . . 8
2.1.4 Orbit mapping, orbit map, and quotient map . . . . . 9
2.1.5 Group convolution . . . . . 9
2.2 Equivariant CNNs on Homogeneous Spaces . . . . . 10
2.3 Fiber lifting and fiber convolutions . . . . . 12

**3 Summary** **15**

**4 Bundle Geodesic Convolutional Neural Network (BGCNN) for DWI Segmentation from Single Scan Learning** **19**
4.1 Abstract . . . . . 19
4.2 Introduction . . . . . 20
4.3 Related work . . . . . 20
4.3.1 Organisation . . . . . 21
4.4 Bundle Geodesic Convolutional Neural Network . . . . . 21

4.4.1 Layer definitions . . . 22
4.4.2 Discretisation and implementation in the case $\mathcal{M} = \mathbb{S}^2$ . 23
4.5 Experiments & Results . . . 23
4.5.1 Experimental setup . . . 23
4.5.2 Results . . . 25
4.6 Discussion and Conclusion . . . 27

**5 Bundle Geodesic Convolutional Neural Network for DWI Segmentation 31**
5.1 Abstract . . . 31
5.2 Introduction . . . 32
5.3 Related work . . . 32
5.4 Method . . . 33
5.4.1 Layer definitions . . . 34
5.4.2 Discretization and implementation in the case $\mathcal{M} = \mathbb{S}^2$ . 36
5.5 Experiments and Results . . . 37
5.5.1 Spinal Scan . . . 37
5.5.2 Synthetic Dataset . . . 39
5.5.3 Human Connectome Brain Scans . . . 40
5.6 Discussion . . . 43
5.7 Conclusion . . . 44

**6 Group Convolutional Neural Network for DWI Segmentation 45**
6.1 abstract . . . 45
6.2 Introduction . . . 45
6.3 Method . . . 46
6.3.1 Standard convolution operations . . . 47
6.3.2 Discretization of spherical signals . . . 48
6.3.3 Generic Networks used in this work . . . 48
6.4 Experiments and Results . . . 49
6.4.1 Experiment setup . . . 50
6.4.2 Results . . . 51
6.5 Conclusion . . . 53

**7 A Study on Group Convolutions and Equivariance for DWI Segmentation 55**
7.1 abstract . . . 55
7.2 Introduction . . . 55
7.3 Related Work . . . 57
7.4 Method . . . 58
7.4.1 Standard convolution operations . . . 58
7.4.2 Pseudo-convolutions on $\mathbb{S}^2$ . . . 60
7.4.3 Discretization of spherical signals . . . 60
7.4.4 Generic Networks used in this work . . . 61

7.5 Experiments and Results . . . . . 63
7.5.1 Experiment setup . . . . . 63
7.5.2 Results . . . . . 65
7.6 Discussion . . . . . 74
7.7 Conclusion . . . . . 74

**8 Discussion and Future Work** **75**

**Bibliography** **77**

## Abstract

CONVOLUTIONAL neural networks (CNN) have been shown to be efficient and effective in image analysis tasks due to their built-in locality and their weight-sharing property. Since convolutions are translation equivariant - a shift in the input results in the same shift in the output - the networks preserve translation symmetry. In fact, for the most common image analysis tasks, the space that the data live in - Euclidean space - is a *principal homogeneous space* to the translation group, it is, consequently, generic and convenient to apply CNNs to this type of data directly. For data modalities that are not measured in spaces like the Euclidean space, nonetheless, the translation equivariance is not immediately satisfied. For example, if we translate a signal on a 2D unit sphere, based on the path that is taken, the resulting signal at the destination will have different orientations. Hence, in this thesis, we focus on generalizing CNNs to more general group actions other than simply translation. We take inspiration from the classical path in literature for generalized CNNs. We first lift the data to groups, and then convolutions are performed on the groups via group actions, after which we project the functions back to the original space to perform tasks. In this sense, the data should be modeled in a way such that it is a function mapping from the homogeneous space of the group it is being lifted to. On the other hand, the group that the data are lifted to is not arbitrary. What kind of actions should be incorporated into the group? In this thesis, we explore the group actions in the most natural way - the actions should be associated with the possible motions that come with the data. In other words, the group action should encode whatever motions the data might have in reality such that the model can capture these motions and thus be resistant to variations in real-world data.

In this thesis, we choose Diffusion Weighted Magnetic Resonance Imaging (DWI) as the data and explore possible group actions that are natural to this type of data. DWI is a technique that captures anisotropies in the movement of molecules in tissues and is very useful in diagnoses of vascular strokes in the brain, among other diagnoses of diseases. It has a structure that differs from regular images - it provides 3-dimensional diffusion information at each voxel that can be encoded as a function on a unit sphere. Therefore, it provides a natural structure for generalized CNNs. The variations in the data - or in other words, symmetries in the data - are 3D rigid motions, which can be easily modeled mathematically, fully, or partly. Unlike existing methods in the literature that use irreducible representations that predefine function basis/filter banks for the spherical convolution, in this thesis, we do not impose predefined functions for the CNN task, and we aim at performing lifting and group convolution in a generic and lightweight way. Instead, the symmetries in the data are reflected by group actions that are the most natural for this type of data. We gradually incorporate more symmetries that are associated with the data and perform a segmentation task. With more symmetries incorporated,

we see a clear increase in the performance of the task. Furthermore, it is shown that the more symmetries are reflected in the modeling, the more resistant the model is to variations in data.

## Dansk Resumé

Konvolutionelle neurale netværk (CNN) har vist sig at være effektive og effektive til billedanalyseopgaver på grund af deres indbyggede lokalitet og deres vægtdelingsegenskab. Da konvolutioner er translationsækvivariante - en skifte i input resulterer i det samme skifte i output - bevarer nettene den translationssymmetri. For de mest almindelige billedanalyseopgaver er det faktisk sådan, at rum, som dataene lever i - det euklidiske rum - er et principielt homogent rum til translationsgruppen, og det er derfor generisk og praktisk at anvende CNN'er på denne type data direkte. For datamodaliteter, der ikke er målt i rum som det euklidiske rum, er translationsekvivariansen ikke desto mindre ikke umiddelbart opfyldt. Hvis vi f.eks. translaterer et signal på en 2D-enhed kugle, baseret på den sti, der tages, vil det resulterende signal på destinationen vil have forskellige orienteringer. I denne afhandling fokuserer vi derfor på at generalisere CNN'er til mere generelle gruppeaktioner end blot translation. Vi tager inspiration fra den klassiske rute i litteraturen for generaliserede CNN'er. Vi først løfter dataene til grupper, og derefter udføres der konvolutioner på grupperne via gruppeaktioner, hvorefter vi projekterer funktionerne tilbage til det oprindelige rum for at gennemføre opgaver. I denne forstand skal dataene modelleres på en sådan måde, at at det er en funktionsafbildning fra det homogene rum i den gruppe, det er bliver løftet til. På den anden side er den gruppe, som dataene løftes til ikke arbitrær. Hvilken slags handlinger skal inkorporeres i gruppen? I denne afhandling undersøger vi gruppens aktioner på den mest naturlige måde - de aktioner bør være forbundet med de mulige bevægelser, der følger med dataene. I andre ord bør gruppeaktionen indkode de bevægelser, som dataene måtte have i virkeligheden, således at modellen kan indfange disse bevægelser og dermed være resistent over for variationer i data fra den virkelige verden. I denne afhandling vælger vi Diffusion Weighted Magnetic Resonance Imaging (DWI) som data og undersøger mulige gruppeaktioner, der er naturlige for denne type data. DWI er en teknik, der indfanger anisotropier i bevægelsen af molekyler i væv, og er meget nyttig i diagnoser af vaskulære slagtilfælde i hjernen, blandt andre diagnoser af sygdomme. Den har en struktur, der adskiller sig fra almindelige billeder - det giver en 3-dimensionel diffusionsinformation ved hver voxel, der kan kodes som en funktion på en enhedssfære. Det giver derfor en naturlig struktur for generaliserede CNN'er. Variationerne i dataene - eller med andre ord symmetrier i dataene - er 3D-rigide bevægelser, som kan let modelleres matematisk, helt eller delvist. I modsætning til eksisterende metoder i litteraturen, der anvender irreducible repræsentationer, der på forhånd definerer funktionsbasis/filterbanker til den sfæriske konvolution, pålægger vi i denne afhandling ikke foruddefinerede funktioner til CNN-opgaven, og vi sigter mod at udføre løft og gruppefoldning på en generisk og letvægts måde. I stedet anvendes symmetrierne i dataene afspejles af gruppehandlinger, der er de mest naturlige for denne type data. Vi inkorporerer gradvist flere symmetrier, der

er forbundet med dataene og udfører en segmenteringsopgave. Med flere symmetrierer taget med, ser vi en klar stigning i opgavens ydeevne. Desuden, det vist, at jo flere symmetrier, der afspejles i modelleringen, jo mere modstandsdygtigere modellen er over for variationer i data.

## Acknowledgements

The past three years have been a life-changing experience for me. The skills and knowledge I have gained will benefit me, I think, for a life-long time. It is both exciting and emotional for me to see it coming to an end.

I would like to start by thanking my mother, who has always been a role model for me. She has taught me to be independent, strong, and responsible, which has always been an irreplaceable inspiration for me through hard times. I would like to thank my father, who has always taken good care of me. These three years would be much harder if my husband, Sveinn, was not going through it with me, always providing me with the strongest support.

I would like to thank my supervisors, François, Kenny, and Sune. They have provided me the help that I needed and the support that is more than I deserved. I would like to especially thank François for equipping me with mathematical skills that are not only useful but also inspirational.

I would like to thank all my friends from DIKU that I made during the three years. They are not only my colleagues but friends that I can share my difficulties and my joys with. I will always cherish the time that we spent together.

At last, I would like to thank the European Union for funding this project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 801199. The papers only contain the author's views. The Research Executive Agency and the Commission are not responsible for any use that may be made of the information it contains.

# Chapter 1

# Introduction

In this chapter, we first introduce the motivation of this thesis, after which we pinpoint the problem properly. Finally, we briefly present the data that are used in this thesis.

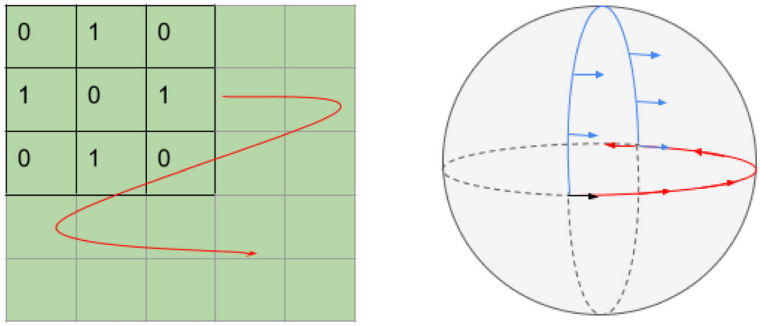
## 1.1 Motivation and Problem Definition

For digital image processing, including medical image processing, machine learning techniques have shown to be robust against variation and noise in data, and convolutional neural networks (CNN) have been shown to be extraordinarily efficient and effective [29, 36]. Convolutions are translation equivariant, meaning that a shift of the input results in the same shift of the output. Therefore, convolutions preserve the translation symmetry. However, for data modalities that do not measure signals in Euclidean spaces, it is not immediately clear how to create equivariant convolutions for the situation. For example, on an oriented manifold, a translation of the input does not generate the same shift in the output, since the path chosen for the translation action is not unique, and different paths can generate different orientations of the output in the destination.

Generally speaking, there are a few challenges in applying CNNs to data in a general sense, since data can be measured in a wide range of spaces, e.g. a non-flat manifold.

### 1.1.1 The underlying space may not have a group structure.

The space that the function lives in poses a challenge for applying CNN to it. In order to define convolution, a criterion must be satisfied - the underlying space must have a group structure or be a homogeneous space of a group. For a Euclidean space $\mathbb{R}^n$ (e.g. $\mathbb{R}^2$ or $\mathbb{R}^3$), this is not a problem since the space is a homogeneous space of the translation group $\mathbb{T}^n$. For data that live on a manifold, e.g. $\mathbb{S}^2$, the translation group does not act on $\mathbb{S}^2$. Translating




(a) Translation group on $\mathbb{R}^2$

(b) Parallel transport on $\mathbb{S}^2$

Figure 1.1: Figure 1.1a shows that the Euclidean space $\mathbb{R}^2$ comes with a translation group structure, thus defining convolution on it is generic and natural. Figure 1.1b shows the path dependency of parallel transport. Moving the black arrow to the opposite side of the sphere ends with different orientations of the arrow due to different paths chosen.

a convolutional kernel on a manifold can be performed by *parallel transport*. There may be no obvious group acting on the manifold, such that the translation of functions on a manifold can be path-dependent, as is illustrated in Figure 1.1. Thus, convolution is not defined in this space. In order to utilize the remarkable efficiency and robustness that CNN has on this type of data structure, special kinds of kernels are required.

Therefore, to deal with this challenge, we must generalize traditional CNNs to more complicated actions than simply translations that are provided by 2D or 3D images. Many works have been done to generalize convolutions to other groups than mere translation [5, 23, 13, 16, 7, 14, 32], either for Euclidean or non-Euclidean data. In this thesis, we take inspiration from the general path of generalizing convolution to group actions provided by Cohen *et al.* [14], with the classical lifting-convolution-projection recipe. In addition, we aim to do this generalization in a direct and natural way, without seeking spectral methods as what is usually done in the literature.

### 1.1.2 Motions that come with the data.

Given the modality of the data, there can be different possible movements that the data can be associated with, e.g. translation, rotation, etc. In a traditional CNN setup for 2D images, only translation motions are incorporated in the convolution. Objects in 2D images can, however, have rotational motions as well, which is usually not taken into account in a regular CNN model.

The most classical approach to deal with this is to use data augmentation,

reflecting the expected symmetries in the data, in the hope that the network will be able to capture the symmetries during the training phase, learning symmetry-aware kernels. This is, however, usually costly either in terms of computation or storage. It was presented by Bekkers *et al.* in [5] that adding this rotational action in the modeling boosts the performance of the CNN without much cost of computation or storage. Therefore, it is, indeed, desirable to incorporate the actions that the data might produce in the modeling, based on the modality of the data.

### 1.1.3 Problem statement

The problems we are trying to solve, consequently, wear down to 1) we want to generalize convolution - thus kernels/filters - to more complex actions for data in general in a direct and natural way; 2) we want to incorporate the motion of the data in the modeling.

In this thesis, we choose Diffusion Weighted Imaging (DWI) to achieve this generalization due to its special data structure and its potential in diagnoses of diseases in general. In the next section, we briefly introduce some basic concepts of DWI.

## 1.2 Data Background: Diffusion Weighted Imaging (DWI)

In this section, we first introduce some basic measurements of the DWI data, after which we present our mathematical interpretation of this type of data such that they can be modeled in a generic way that reflects the structure of the data.

### 1.2.1 DWI Basics

In its original terms, diffusion is the microscopic movement of atoms and molecules in solution and gas. In human bodies, molecules of water, salt, and various kinds of chemicals flow freely among living tissues. Diffusion Weighted Imaging (DWI) is a form of Magnetic Resonance Imaging (MRI) based on measuring movements of molecules within a voxel of tissue, and the resulting signals are anisotropic - it is not directionally uniform. It uses specific MRI sequences to generate contrast in MR images. To measure the significance of diffusion, the diffusion coefficient ($D$) is used. The diffusion coefficient is the quantity of a substance that, in diffusing from one region to another, passes through each unit of cross-section per unit of time when the volume-concentration gradient is unity [33]. For tissues that are highly cellular or tissues with cellular swelling, the diffusion coefficients are lower. DWI has an essential role in diagnosing ischemia and vascular strokes in the brain as well as characterizing tumors, such that there is a strong incentive to automate the

analysis of this type of data. Therefore, efficiently and informatively modeling DWI data is crucial and inevitable for aiding the next steps of diagnosing diseases.

To measure signals in terms of diffusion, several parameters are used. To explain and show how these parameters affect the output, in the following text of this section, we introduce the two most important concepts: $b$-value and $b$-vector.

**$b$-value**: To generate MRI contrast, specific pulse sequences and intrinsic tissue properties are used, in which the parameters are adjusted. The contrast is based on weighted properties of tissues [17]. The factor to weigh diffusion-sensitizing is called the $b$-value:

$$b = \gamma^2 \cdot G^2 \cdot \delta^2(\Delta - \frac{\delta}{3}) \tag{1.2.1}$$

where $\gamma$ is the gyromagnetic ratio, $G$ is the strength of the diffusion-sensitizing gradients, $\delta$ is the duration of the gradient pulse, and $\Delta$ is the time interval between these gradients. The unit of $b$ is $sec/mm^2$. In practice, $b$ is increased by increasing the amplitude ($G$) and duration ($\delta$) of the gradient.

The higher $b$ is, the stronger the diffusion affects the signals. Suppose $S_0$ is the baseline MR signal, and $D$ is the diffusion coefficient. Then the signal $S$ after applied with the diffusion gradients will be:

$$S = S_0 \cdot e^{-bD} \tag{1.2.2}$$

**$b$-vector**: Within a voxel of tissue, the $b$-vectors correspond to the directions of the diffusion sensitivity. Thus, a voxel contains signal intensities $S$ in the directions indicated by the $b$-vectors. For a given $b$-value, a set of $b$-vectors is provided for the scan shared by all voxels.

An example of a diffusion image can be found in Figure 1.2. For a given $b$-value, the signals are measured in different directions ($b$-vectors), each direction corresponds to a volume image.

A DWI scan with a single $b$-value is named a *single-shell image*, and a scan with multiple $b$-values is called *multi-shell image*. Furthermore, the signals defined in these directions in a voxel have antipodal symmetry - the opposite direction of a $b$-vector has the same signal value as the $b$-vector.

### 1.2.2 Mathematical Interpretation of DWI

As explained above, for a given $b$-value (parameterized by the magnitude, duration, and interval of the gradients), there is a set of $b$-vectors associated with the $b$-value indicating directions of diffusion sensitivity for each voxel in a scan, and all voxels share the same set of $b$-vectors. A direction in the 3D Euclidean space $\mathbb{R}^3$ can be seen as a point on the surface of a unit sphere, thus a voxel of a DWI image consists of signal scalar values defined on some

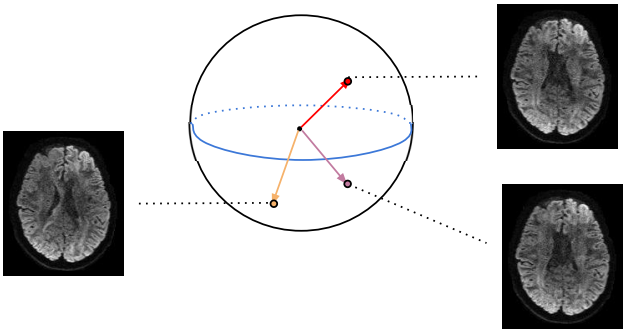


Figure 1.2: Illustration of the same slice of a DWI scan from different directions.

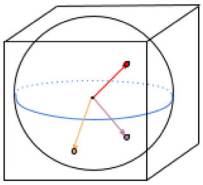


Figure 1.3: Illustration of a voxel in a DWI scan of a given $b$-value. A set of directions ($b$-vectors) is provided, and signal values (scalars) are defined in these directions.

points ($b$-vectors) on a unit sphere. An illustration of this interpretation can be found in Figure 1.3.

Furthermore, for a given $b$-value, we can generalize each voxel in a DWI scan image to a continuous function defined on a unit sphere surface $\mathbb{S}^2$ with some interpolation scheme that preserves the antipodal symmetry. For this purpose, we use the Watson Kernel [26] in this thesis. Therefore, each voxel is a function $I_{\boldsymbol{x}} : \mathbb{S}^2 \to \mathbb{R}$ at voxel location $\boldsymbol{x}$. Taking all the voxels into account, a *single-shell* DWI scan image is a function $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. Without loss of generality, a *multi-shell* DWI image is a function $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}^N$, where $N$ is the number of $b$-values used in the scan. In this thesis, we only look at the *single-shell* case since it is the most common type of data.

# Chapter 2

# CNN Background: Geometric Deep Learning and Equivariant Neural Networks

In this chapter, we showcase the most commonly used setup of CNNs in image analysis and generalize it to a group action fashion. In literature, the generalization of convolutions on spheres is usually done by *irreducible representations*. We take a different path than that so that the lifting of functions to groups and the generalized convolutions can be performed in a lightweight and direct way. We build the generalized convolutions on homogeneous spaces of groups to serve as a mathematical foundation of this thesis.

## 2.1 Generalizing Convolution to Group Actions

We follow the convention in the machine learning society of viewing correlation as convolution. Here we showcase the 2D case for simplicity, and it can be easily generalized to higher dimensions. A 2D convolution on a function $h : \mathbb{R}^2 \to \mathbb{R}^n$ can be written as:

$$(\kappa * h)(x, y) = \int_{x\in\mathbb{R}} \int_{y\in\mathbb{R}} \kappa(x', y')h(x - x', y - y')dx'dy', \tag{2.1.1}$$

where $\kappa : \mathbb{R}^2 \to \mathbb{R}^n$ is a function to convolve with $h$, and is usually referred to as a kernel. The 2D translation group $\mathbb{T}^2$ does act on the space of images on the left (see Section 2.1.1 for details of actions on functions):

$$\tau_{\vec{v}}h(x, y) = h(x - x', y - y'), \quad \tau_{\vec{v}} \in \mathbb{T}^2, \quad \vec{v} = (x', y')^T \tag{2.1.2}$$

Therefore, Equation (2.1.1) can be rewritten in a group theoretical way as:

$$(\kappa * h)(x, y) = \int_{x\in\mathbb{R}} \int_{y\in\mathbb{R}} \kappa(x', y')(\tau_{\vec{v}}h)(x, y)dx'dy'. \tag{2.1.3}$$

Before continuing, we introduce some classical definitions in group theory that are crucial to the later sections.

### 2.1.1 $G$-action, Homogeneous spaces

Given a space $\mathcal{M}$ and a group $G$ with identity element $e$, a left-action of $G$ on $\mathcal{M}$, denoted by $\cdot$, is a function $G \times \mathcal{M} \to \mathcal{M}$ that satisfies:

- Identity: $e \cdot m = m, \forall m \in \mathcal{M}$.
- Compatibility: $g \cdot (g' \cdot m) = (gg') \cdot m$, for $g, g' \in G, m \in \mathcal{M}$.

With the two axioms satisfied, it follows that for all $g \in G$, the mapping $g \cdot \mathcal{M} \to \mathcal{M}$ is a bijection. For an $m \in \mathcal{M}$, the *orbit* $G(m)$ is the set $\{g \cdot m, g \in G\}$. The *stabilizer* $G_m$ of $m$ is the set of transformations that keep $m$ fixed: $G_m = \{g \in G, g \cdot m = m\}$, and it is a subgroup of $G$. There is a trivial but important property of stabilizers: for all pairs $m_1, m_2$ on the same orbit, i.e., there exists $g \in G$ s.t. $m_2 = g \cdot m_1$, then their stabilizers are conjugated $G_{m_2} = gG_{m_1}g^{-1}$. $\mathcal{M}$ is a *homogeneous space* of $G$ if it consists of only one orbit: for all $m_1, m_2 \in \mathcal{M}$, there exists a $g \in G$, s.t. $m_2 = g \cdot m_1$.

#### 2.1.1.1 $G$-action on functions

If $\mathcal{M}$ is endowed with a left $G$-action, then a vector space of functions $f : \mathcal{M} \to \mathbb{R}^N$ is endowed with the left $G$-action, often denoted by $L_g$: $L_g f(m) = f(g^{-1} \cdot m)$ for all $g \in G$ and all $m \in \mathcal{M}$. In general, we consider the function spaces $L^2(\mathcal{M}, \mathbb{R}^n)$ for a $G$-invariant measure on $\mathcal{M}$.

### 2.1.2 Equivariant Map

Assume that $\mathcal{M}$ and $\mathcal{N}$ are sets, both endowed with a $G$-action. A map $f : \mathcal{M} \to \mathcal{N}$ is equivariant if $f(g \cdot m) = g \cdot f(m)$ for all $g \in G$ and all $m \in \mathcal{M}$.

### 2.1.3 Quotient space

The quotient of $G$ by a subgroup $H$ partitions $G$ into translated copies of $H$: $G/H = \{gH, g \in G\}$. Here $gH$ is called the *coset* of $g$. The map that sends $g$ to $gH$ is called the *projection* onto $G/H$.

### 2.1.4 Orbit mapping, orbit map, and quotient map

Assume $\mathcal{M}$ is a homogeneous space of $G$. Take $m_0 \in \mathcal{M}$ and $H = G_{m_0}$ its stabilizer. There is a commutative diagram

$$\begin{array}{ccc} G & \xrightarrow{\ell_{m_0} = \cdot m_0} & \mathcal{M} \\ \downarrow{\scriptstyle \pi} & \nearrow & \\ G/H & & \end{array} \tag{2.1.4}$$

The horizontal arrow is the *orbit mapping*, and $\pi$ is the *quotient map*. The diagonal arrow is the *orbit map*, and it is a diffeomorphism. Now $\ell_{m_0}^{-1}(m) = \{g \in G, g \cdot m_0 = m\} = gH$ for any of these $g$s. Here $gH$ is the *fiber* over $m$. Changing $m_0$ will change the orbit map up to a unique diffeomorphism.

Here we give a simple but important example. Take the $SO(3)$ group as $G$ and $\mathbb{S}^2$ as $\mathcal{M}$. $\mathbb{S}^2$ is obviously a homogeneous space of $SO(3)$. Choose $x_0 \in \mathbb{S}^2$, and $H = SO(3)_{x_0}$ its stabilizer, then the orbit map $SO(3)/SO(3)_{x_0} \cong \mathbb{S}^2$ is a diffeomorphism. It is easily seen as well that $SO(3)_{x_0}$ is diffeomorphic to $SO(2)$.

### 2.1.5 Group convolution

Revisiting the convolution in Equation (2.1.3), it is clearly equivariant with respect to translation:

$$(\kappa * (\tau_{\vec{v}} h))(x, y) = (\tau_{\vec{v}}(\kappa * h))(x, y). \tag{2.1.5}$$

$\mathbb{R}^2$ is, of course, a homogeneous space of $\mathbb{T}^2$, since for all $p, q \in \mathbb{R}^2$, there exists $\tau_p \in \mathbb{T}^2$, s.t. $\tau_p \cdot p = q$. Additionally, $\mathbb{R}^2$ is diffeomorphic to $\mathbb{T}^2$, $\mathbb{R}^2 \cong \mathbb{T}^2$, thus it is itself a translation group as well. In this case, $\mathbb{R}^2$ is actually a *principal homogeneous space* of $\mathbb{T}^2$. The general feature representation $\mathbb{R}^n$ (here we use $\mathbb{R}^n$ instead of $\mathbb{R}^{N_c}$ for simplicity and generalization purpose) is a vector space.

Therefore, we write this recipe in a more generalized way - in the group theoretical sense. We replace the translation group $\mathbb{R}^2$ with a arbitrary group $G$ and the feature space with a vector space $V$, the feature map can be generalized to:

$$f : G \to V, \tag{2.1.6}$$

and Equation (2.1.3) can be rewritten as:

$$(\boldsymbol{k} * f)(g) = \int_G \boldsymbol{k}(g^{-1}h) f(h) dh, \tag{2.1.7}$$

where $dh$ is a left-invariant Haar measure on $G$ [18].

The convolution by a generalized kernel $f \mapsto \boldsymbol{k} * f$ is a function that is $G$-equivariant: $\boldsymbol{k}* \in Hom_G(L^2(G,V), L^2(G,W))$. $W$ is some vector space, not necessarily the same dimension as $V$. For example, the dimensions of $V$ and $W$ can be seen as different input and output channels of a layer in a CNN. The feature map of the generalized convolution Equation (2.1.7) can be written as:

$$(\boldsymbol{k} * f) : G \to W. \tag{2.1.8}$$

This equivariance generalizes Equation (2.1.5), this time with respect to the left group action:

$$(\boldsymbol{k} * (L_g f)) = L_g(\boldsymbol{k} * f) \tag{2.1.9}$$

We refer the readers to [23] for more detailed theoretical foundation.

## 2.2 Equivariant CNNs on Homogeneous Spaces

Many works have contributed to the generalization of CNNs to *Group Equivariant* CNNs (GCNNs) on Euclidean spaces and spheres [13, 16, 5, 28, 48, 15, 47].Few works are focusing on building equivariant networks for data with a different modality, e.g. DWI. In Mueller *et al.* [34], they proposed a roto-translational equivariant network for diffusion MRI using filter banks developed from spherical harmonics and radial basis. Yet there is no work in literature that built equivariant CNNs for this type of data modality in the most generic way without predefined basis. In fact, it is not clear how to do so without a general theory of GCNNs on homogeneous spaces.

In Equation (2.1.7), the convolution is performed on a group $G$. For images $\tilde{f} : \mathbb{R}^2 \to \mathbb{R}^3$ in the traditional CNN case, $\mathbb{R}^2$ being a principal homogeneous space of $\mathbb{T}^2$ gives us the privilege to rewrite the image function as $\tilde{f} : G \to \mathbb{R}^3$, and here $G \cong \mathbb{R}^2$.

This is not always the case for other types of data, e.g. non-flat data like in DWI. For functions defined on an arbitrary space, e.g. a manifold, $\mathcal{M}$, the space itself does not necessarily have a group structure, so the condition for defining convolution is not satisfied. To deal with this, the usual way is to lift the function in a convolutional fashion from $\mathcal{M}$ to a specific group. The convolutional lifting is defined as follow:

$$(\bar{\boldsymbol{k}} * \bar{f})(g) = \int_{\mathcal{M}} \bar{\boldsymbol{k}}(g^{-1}m)\bar{f}(m)dm, g \in G. \tag{2.2.1}$$

Here the group $G$ is not arbitrary. The convolutional lifting is performed on $\mathcal{M}$, which does not have a group structure. Thus to do convolution on it, it must be a homogeneous space of the group $G$ it is lifted onto. In this way, a function $f : \mathcal{M} \to \mathbb{R}^n$ is lifted to a function $f\uparrow$: $G \to \mathbb{R}^n$

$$f\uparrow (gH) \equiv f(m) \tag{2.2.2}$$

the lifting is constant on the cosets of $G/H$.

For more solid and general foundations of equivariant CNNs on homogeneous spaces, we refer the readers to [14].

After the lifting of data functions to group structures, group convolutions are performed using Equation (2.1.7). At the end of a network, the functions are usually projected back to the original space of the input data in order to perform tasks for the model.

From a feature map $F : G \to \mathbb{R}^n$, a projection will provide an associated map $\overline{F} : \mathbb{M} \to \mathbb{R}^n$. Using the orbit map from (2.1.4), one integrates fiber-wise: *(nonlinear) averaging*, min or max-pooling for instance. Here we demonstrate the case for the nonlinear averaging, min and max pooling can be written as limiting cases.

$$\forall g, g \cdot m_0 = m, \quad \overline{F}(m) = \frac{1}{|H|} \int_H F(gh)\, dh. \tag{2.2.3}$$

When $G$ is a finite dimensional Lie group, and with some properness on the action, then $H$ will be compact, thus with finite measure such that it can be normalized to 1. So we can assume $|H| = 1$. The projection keeps the equivariance property such that attaching it after the lifting and convolutional layers does not break the equivariance.

**Lemma 2.2.1** *$\overline{F}$ is equivariant: $\overline{L_k F} = L_k \overline{F}$.*

**Proof.** Pick a $g$ such that $g \cdot m_0 = m$, Then for any $h \in H$, $k^{-1}gh \cdot m_0 = k^{-1}m$.

$$\begin{aligned} \overline{L_k F}(m) &= \int_H L_k F(gh)\, dh \\ &= \int_H F(k^{-1}gh)\, dh = \overline{F}(k^{-1}m) = L_k \overline{F}(m) \end{aligned}$$

Therefore, a generalized group convolutional neural network can be established using the following recipe:

- Step 1: Lifting using Equation (2.2.1)
- Step 2: Group convolutions using Equation (2.1.7)
- Step 3: Projection (pooling) using Equation (2.2.3)

With the generalization of CNNs to groups and homogeneous spaces, we can perform CNNs with more complicated group actions (e.g. rotation) than just translation as in image analysis works that use traditional CNNs. Lifting functions to groups enables a large range of interactions of local geometry in data based on the actions of the group.

## 2.3 Fiber lifting and fiber convolutions

In this thesis, before lifting the functions to the full group $SO(3)$ of the manifold $\mathbb{S}^2$, we first experimented with lifting the spherical functions locally to the stabilizer of each point on the manifold, for a general manifold setup. Now, we explain this kind of lifting and the convolutions that come with this kind of lifting.

#### 2.3.0.1 Lifting

Here let $m_0 \in \mathcal{M}$ be our *base point* and $\kappa_{m_0} \in L^2\left(T_{m_0}\mathcal{M}\right)$. Choose, for each $m \in \mathcal{M}$, a transformation $\sigma_m$, among these for which $g \cdot m_0 = m$ (i.e. the coset $gH$). With such a choice, $G_m = \sigma_m H \sigma_m^{-1}$. Here we use the fact that the action of $g$ on $\mathcal{M}$ gives rise to a tangent action isomorphism $L_g = T_g : T_m\mathcal{M} \to T_{g\cdot m}\mathcal{M}$, and it is still denoted by $g\cdot$.

The mapping $\sigma : \mathcal{M} \to G$ is a *section* of the orbit mapping $\ell_{m_0}$. Finding a smooth global map $\sigma$ might not be possible. Our fundamental example is $\mathbb{S}^2$ as $SO(3)$-homogeneous space, and $m_0$ is in $\mathbb{S}^2$, the "north pole" $\mathbf{e}_3 = (0,0,1)^T$, for instance. In differential geometry, it is well-known that there is no smooth global section $\sigma : \mathbb{S}^2 \to SO(3)$.

**Remark 2.3.1** *Still, one can define such a $\sigma$ on $\mathbb{S}^2\backslash\{-m_0\}$ easily via the cross product. Use the isomorphism*

$$\mathbb{R}^3 \xrightarrow{\widehat{\ }} \mathfrak{so}(3), \quad \mathbf{u} \mapsto \{\hat{\mathbf{u}} : \mathbf{v} \mapsto \hat{\mathbf{u}}\mathbf{v} = \mathbf{u} \times \mathbf{v}\}. \tag{2.3.1}$$

*For $q \in \mathbb{S}^2\backslash\{m_0, -m_0\}$, set $\mathbf{l}_q = \frac{m_0 \times q}{|m_0 \times q|}$, $\theta = \arccos m_0^\top q$ and $R_q = e^{\theta \hat{\mathbf{l}}_q}$. For $q = m_0$, set $R_{m_0} = \mathrm{id}$. Then $R_q m_0 = q$ and $q \mapsto R_q$ is clearly smooth.*

Back to the general definition. Choose $\sigma$ a section $\mathcal{M} \to G$, then for each $q \in \mathcal{M}$, one can *lift* $f$ to $G_q$ by defining, for $g \in G_q$,

$$\left(\kappa \star_\sigma f\right)_q (g) = \int_{T_q\mathcal{M}} f(\mathrm{Exp}_q\, v)\sigma_q \cdot \kappa_{m_0}(g^{-1}v)\, dv \tag{2.3.2}$$

The transformation $\sigma_q$ acts on $\kappa_{m_0}$ as $\sigma_q \cdot \kappa_{m_0} : v \in T_q\mathcal{M} \mapsto \kappa_{m_0}\left(\sigma_q^{-1}v\right)$. Since $g \in G_q$, $g$ acts by isomorphism on $T_q\mathcal{M}$ such that $g^{-1}v \in T_q\mathcal{M}$.

In the case of interest, $\mathcal{M} = \mathbb{S}^2$, $G = SO(3)$, and $G_q = SO(T_q\mathbb{S}^2)$. Equivariance of the lifting would require the kernel to be invariant under rotations, otherwise $\sigma_{Rq} = R\sigma_q$ must be satisfied for $R \in SO(3)$. This, in general, does not hold. Therefore, unless severe restrictions are imposed on the kernel, the lifting does not have equivariance.

#### 2.3.0.2 Fiber-convolution.

In the sequel $\mathcal{M} = \mathbb{S}^2$, one has a *feature map* $F : \mathcal{SO}(\mathcal{S}^2) \to \mathbb{R}^N$. At a given point $m_0 \in \mathcal{S}^2$, let $K_{m_0} : SO(T_{m_0}\mathbb{S}^2) \to \mathbb{R}$ be a kernel on the fiber $SO(T_{m_0})$. Such a kernel gives rise to a family of kernels on each fiber: at a $q \in \mathbb{S}^2$, find $R \in SO(3)$ with $Rm_0 = q$ and set $K_q : S \in SO(T_q\mathbb{S}) \mapsto K_{m_0}(R^T SR)$. This is actually independent of the choice of a translation $R : m_0 \to q$, as opposed to the lifting case, where we used a section $\sigma$ of the orbit mapping.

**Lemma 2.3.2** *Let $R$ and $R'$ be two rotations which translate $m_0$ to $q$ and $S \in SO(T_q\mathbb{S}^2)$. Then $R^T SR = R'^T SR'$.*

**Proof.** Two such rotations differ by an element $Q \in SO(T_{m_0}\mathbb{S}^2)$, $R' = RQ$. So if $\Sigma = R^T SR \in T_{m_0}\mathbb{S}^2$, $R'^T SR' = Q^T \Sigma Q \in T_{m_0}\mathbb{S}^2$ is conjugate to $\Sigma$. $SO(2)$ is Abelian, therefore, conjugation is trivial. □

Now we can defined the "Fiber-convolution" of $F : \mathcal{SO}(2) \to \mathbb{R}^N$ with $K_{m_0} : L^2(T_{m_0}\mathcal{S}^2)$ by

$$(K \star F)(q, S) = \int_{SO(T_q\mathbb{S}^2)} F_q(T) K_{m_0}(\left(R^\top TR\right)^\top S)\, dT$$

for any $R$ translating $m_0$ to $q$.

#### 2.3.0.3 Projection - Fiber Collapse

A feature $F : \mathcal{SO}(\mathcal{S}^2)$ can be collapsed fiber-wise to a $\mathbb{S}^2$-function in many ways, for instance

$$\begin{aligned} \mathbf{f}(q) &= \max_{S \in SO(T_q\mathcal{S}^2)} F_q(S), \\ \mathbf{f}(q) &= \min_{S \in SO(T_q\mathcal{S}^2)} F_q(S), \\ \mathbf{f}(q) &= \int_{SO(T_q\mathcal{S}^2)} F_q(S)\, dS \end{aligned} \tag{2.3.3}$$

# Chapter 3

# Summary

In this chapter, we summarize the contributions of all the publications that we produced. This thesis is a compilation of the following works:

- Bundle Geodesic Convolutional Neural Network (BGCNN) for DWI Segmentation from Single Scan Learning. This work was accepted in Computational Diffusion MRI 2021, and it is presented in Chapter 4.
- Bundle Geodesic Convolutional Neural Network for DWI Segmentation. This work was submitted to The Journal of Medical Imaging, and it is under review. It is presented in Chapter 5.
- Group Convolutional Neural Network for DWI Segmentation. This work was submitted to MICCAI 2022, and it is under review. It is presented in Chapter 6.
- A Study on Group Convolutions and Equivariance for DWI Segmentation. This work was submitted to IEEE Transactions in Medical Imaging, and it is under review. It is presented in Chapter 7.

## Summary of Contributions

In Chapter 4, we present a Bundle Geodesic Convolutional Neural Network (BGCNN) where the convolutional kernels are defined on the tangent spaces of $\mathbb{S}^2$. To deal with the unusual space that DWI data are defined on: $\mathbb{R}^3 \times \mathbb{S}^2$, in this work, we first discard the Euclidean part of the data structure and only look at one voxel at a time. In this way, the problem wears down to classifying spherical functions. As was introduced in Section 2.3, instead of lifting a signal $f : \mathbb{S}^2 \to \mathbb{R}$ to $SO(3)$, we lift it, above each point $x$ of $\mathbb{S}^2$, to the the stabilizer $SO_x(2)$ of $x$, via a set of localized spherical kernels transported along predetermined paths above each point where we analyze our signal. The lifted space is isomorphic to $\mathbb{S}^2 \times SO(2)$. Then convolutions are performed on

each fiber, independently of each other. A local rotation pooling is performed to project the information back on $\mathbb{S}^2$, before being fed to a fully convolutional network for classification. The use of these predetermined transports breaks $SO(3)$-equivariance. We compare the proposed method with a multi-layer perceptron using the human connectome project (HCP) DWI dataset [46], and experiments show that our method produces promising results with a very limited number of parameters and a very small set of training data.

In Chapter 5, we provide a detailed analysis of the BGCNN proposed in Chapter 4. In this paper, we compare the proposed method with the state-of-the-art [15] and a multi-layer perceptron using 3 datasets: a DWI scan conducted on a spinal cord, a synthetic dataset that was generated by us, and the HCP DWI dataset [46]. The experiments show that our method achieves the same level of performance as [15] while using models with far lower capacity. In addition, a model sensitivity analysis is conducted for our method, in which a proportion of the training set is used for learning, and the test set remains the same. It is shown from the sensitivity analysis that the performance of the model decreases mildly with the reduction of the training set, but it provides a good trade-off for dealing with the class imbalance in the dataset. Furthermore, it is shown from the analysis that the potential in aiding manual data annotation using our method - a clinician only has to label a part of a scan and provide the labels to our model, the rest of the labeling can be automated by our model.

In Chapter 6, we take a step further from BGCNN, bringing back the $\mathbb{R}^3$ part of the DWI data, and we build convolutions for data in $\mathbb{R}^3 \times \mathbb{S}^2$. Again, the HCP DWI dataset [46] is used. For DWI data, a rigid transformation of a sample, i.e. by the action of the group $SE(3)$, should be reflected, up to the limitations of acquisition protocol, in the signal. The space $\mathbb{R}^3 \times \mathbb{S}^2$ is a *homogeneous space* under the action of $SE(3)$: a point in $\mathbb{R}^3 \times \mathbb{S}^2$ can be transformed in any other point by a rigid transformation. Therefore, we propose a group convolutional neural network that incorporates this action, which is a natural action for this type of data. The $SE(3)$-GCNN we propose encodes the interplay between the spatial and directional parts of the data. It shows robustness in performance and shows resistance to data variation compared to methods that do not encode this type of interaction.

In Chapter 7, we provide a detailed ablation study of a series of equivariant networks built on DWI data $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ using the HCP DWI dataset [46]. We showcase the importance of the full equivariance we imposed in the network, in which the rotation group actions in both parts ($\mathbb{R}^3$ and $\mathbb{S}^2$) of the product space ($\mathbb{R}^3 \times \mathbb{S}^2$) are fully aligned - it is an $SE(3)$-GCNN. We present cases where the rotation group actions in $\mathbb{R}^3$ and $\mathbb{S}^2$ are 1) decoupled, 2) partly aligned, 3) fully aligned, along with extreme cases where there is no

spatial information accounted - an $SO(3)$-GCNN, and no spherical information accounted - a classical CNN in which we discard the geometric structure in voxels completely. From the comparisons, tested with both original and synthetically transformed data, it is shown that networks that have both spatial ($\mathbb{R}^3$) and directional ($\mathbb{S}^2$) information taken into account show the most robust performance. Moreover, the experiments show, however, that models with fully aligned rotation actions encoded in $\mathbb{R}^3$ and $\mathbb{S}^2$ do not perform better than models with decoupled rotation actions. They are, though, the most resistant to variations in data. Therefore, the $SE(3) - GCNN$ demonstrates great potential in real-world situations.

# Chapter 4

# Bundle Geodesic Convolutional Neural Network (BGCNN) for DWI Segmentation from Single Scan Learning

Renfei Liu, François Lauze, Kenny Erleben, Sune Darkner

## 4.1 Abstract

We present a tissue classifier for Magnetic Resonance Diffusion Weighted Imaging (DWI) data trained from a single subject with a single b-value. The classifier is based on a Riemannian Deep Learning framework for extracting features with rotational invariance, where we extend a G-CNN learning architecture generically on a Riemannian manifold. We validate our framework using single-shell DWI data with a very limited amount of training data - only 1 scan. The proposed framework mainly consists of three layers: a lifting layer that locally represents and convolves data on tangent spaces to produce a family of functions defined on the rotation groups of the tangent spaces, i.e., a *section* of a bundle of rotational functions on the manifold; a group convolution layer that convolves this section with rotation kernels to produce a new section; and a projection layer using maximisation to collapse this local data to form new manifold based functions. We present an instantiation on the 2-dimensional sphere where the DWI orientation data is in general represented, and we use it for voxel classification. We show that this allows us to learn a classifier for cerebrospinal fluid (CSF) - subcortical - grey matter - white matter classification from only one scan.

## 4.2 Introduction

Very little manually annotated DWI data exists, and DWI studies are, in general, small in sample size. This poses a challenge for machine learning techniques that for most parts require a significant amount of training data. However, learning from only 1 single-shell scan is possible by constructing a G-CNN architecture that takes advantage of the geometry of the data. This work focuses on building a neural network (NN) for data on manifolds with some form of orientation invariance, and here we take Diffusion Weighted Imaging as the main application. Our goal is to be able to understand spherical patterns up to rotations. There are series of proposals to generalise a $\mathbb{R}^2$ convolutional neural network to curved spaces. In general, to define convolution, the underlying space must have a group structure or be a homogeneous space of a group. This is not always the case for curved space. But even when it is, this often imposes a certain type of filters. In our case, rotational invariance is a desirable property we want in the design. We propose a general architecture for extracting and filtering local orientation information of data defined on a manifold. The architecture allows us to learn similar orientation structures which can appear at different locations on the manifold. Reasonable manifolds have local orientation structures – rotations on tangent spaces. Our architecture lifts data to these structures and performs local filtering on them, before collapsing them back to obtained filtered features on the manifold. This provides both rotational invariance and flexibility in design, without having to resort to complex embeddings in Euclidean spaces. We provide an explicit construction of the architecture for DWI data and show very promising results for this case including single scan learning.

## 4.3 Related work

The importance of the extraction of rotationally invariant features beyond Fractional Anisotropy [4] has been recognized in series of DWI works. [8] developed invariant polynomials of spherical harmonic (SH) expansion coefficients, and discussed their application in population studies. [38] proposed a related construction using eigenvalue decomposition of SH operators. [35] and [52] argued their usefulness for understanding microstructures in relation to DWI.

There is though a vast growth in literature on Deep Learning (DL) for non-flat data or more complex group actions than just translations. [32] proposed a NN on surfaces that extracts local rotationally invariant features. A non-rotationally invariant modification was proposed in [7]. On the other hand, convolution generalises to more group actions than just translation, and this has led to group-convolution neural networks for structures where these operations are supported, especially Lie groups themselves and their homogeneous

spaces [22, 16, 47, 50, 28, 5, 1]. Global equivariance is often sought but proved complicated or even elusive in many cases when the underlying geometry is non-trivial [44]. An elementary construction on a general manifold is proposed in [37] via a fixed choice of geodesic paths used to transport filters between points on the manifold, ignoring the effects of path dependency (holonomy). Removing this dependency can be obtained by summarising local responses over local orientations, this is what is done in [32]. To explicitly deal with holonomy, [42] proposed a convolution construction on manifolds based on stochastic processes via the frame bundle, but it is at this point still very theoretical.

A few works have applied DL to DWI. [24] built multi-layer perceptrons in $q$-space for kurtosis and NODDI mappings. Because of the spherical structure of the DWI data and the homogeneous structure of the sphere, [9] proposed an rotation equivariant construction inspired by [15] for disease classification. [34] propose a sixth-D, 3D space and $q$-space NNs with roto-translation / rotation equivalence properties.

In this work, we are interested in rotationally invariant features, so we take a path closer to [37, 32], but we add an extra local group convolution layer before summarising the data and eliminating path dependency. The proposed construction thus applies to oriented Riemannian manifolds, and no other structure (e.g. homogeneous or symmetric space) is used.

### 4.3.1 Organisation

We introduce the construction in the next section, first in a general setting, then in our case of interest, the sphere $\mathbb{S}^2$. We present experiments and results in section Section 4.5. Discussion and conclusion are presented in section Section 4.6.

## 4.4 Bundle Geodesic Convolutional Neural Network

Bekkers *et al.* [5] used the fact that $SE(2)$ acts on $\mathbb{R}^2$ to lift 2D (vector-values) images to $\mathbb{R}^2 \times \mathbb{S}^1$ via *correlation kernels*. This is not in general the case when $\mathbb{R}^2$ is replaced by an oriented Riemannian manifold $\mathcal{M}$ as there is no roto-translation group defined on a general manifold. An alternative construction is however possible by combining [5] and [32], to obtain a 3-component layer architecture: i) the **lifting layer**, ii) the **group correlation layer**, and iii) **the projection layer**. In practical applications, one or more of these multilayers can be used and a fully connected layer is built upon the last one. In this section, we focus only on the Riemannian part.

We refer the readers to [19] for the Riemannian geometric constructions. In the sequel, a base point $\boldsymbol{x}_0$ is chosen on $\mathcal{M}$. A piecewise smooth path $\gamma$

joining $\boldsymbol{x}_0$ and $\boldsymbol{x} \in \mathcal{M}$ is a continuous curve that may fail to be smooth at a finite number of points. With such a curve, there is a *parallel transport* $P_\gamma$ between $T_{\boldsymbol{x}_0}\mathcal{M}$ and $T_{\boldsymbol{x}}\mathcal{M}$. This is an orientation preserving isometry between tangent spaces. A tangent *kernel* at $\boldsymbol{x}_0$ is a function $\kappa : T_{\boldsymbol{x}_0}\mathcal{M} \to \mathbb{R}^N$. We assume it has a "small support". A rotational kernel at $\boldsymbol{x}_0$ is a function $K : SO(\boldsymbol{x}_0) \to \mathbb{R}^M$, where, $SO(\boldsymbol{x})$ denotes the rotation group of $T_{\boldsymbol{x}}\mathcal{M}$.

### 4.4.1 Layer definitions

As it is usually the case that correlation replaces convolution in convolutional neural networks (CNN). The first two layers will be defined via correlation.
**Lifting layer.** The correlation $f\tilde{\star}_\gamma\kappa$ of $f \in L^2(\mathcal{M}, \mathbb{R}^N)$ is defined as the function on $SO(\boldsymbol{x})$

$$f\tilde{\star}_\gamma\kappa(S) = \sum_{i=1}^{N} \int_{T_{\boldsymbol{x}}\mathcal{M}} \kappa_i(P_\gamma^{-1}S^{-1}v) f_i(\mathrm{Exp}_{\boldsymbol{x}}(v))\, dv \tag{4.4.1}$$

We assume that $\kappa \circ P_\gamma^{-1}$, the support of $\kappa$, is sufficiently small so that the exponential map is injective. For any other path $\delta$ between $\boldsymbol{x}_0$ and $\boldsymbol{x}$, it is easy to show that there exists a rotation $R \in SO(T_{\boldsymbol{x}}\mathcal{M})$ that only depends on $P_\gamma$ and $P_\delta$ with $f\tilde{\star}_\delta\kappa(S) = f\tilde{\star}_\gamma\kappa(RS)$. For any point $\boldsymbol{x}$ and a path $\gamma_{\boldsymbol{x}}$ between $\boldsymbol{x}_0$ and $\boldsymbol{x}$, this filters/lifts $f$ to functions $F_{\boldsymbol{x}} : SO(\boldsymbol{x}) \to \mathbb{R}$. Using an input $f^{(\ell-1)} : \mathcal{M} \to \mathbb{R}^{N_{\ell-1}}$ and $N_\ell$ $\boldsymbol{x}_0$-kernels $\boldsymbol{\kappa}^{(\ell)} = \left(\kappa_1^{(\ell)}, \ldots, \kappa_{N_\ell}^{(\ell)}\right)$, $\kappa_i^{(\ell)}$ $\mathbb{R}^{N_{\ell-1}}$-valued at layer $\ell - 1$,

$$\forall \boldsymbol{x} \in \mathcal{M}, \quad F_{\boldsymbol{x}}^{(\ell)} = \left(f^{(\ell-1)}\tilde{\star}_{\gamma_{\boldsymbol{x}}}\kappa^1, \ldots, f^{(\ell-1)}\tilde{\star}_{\gamma_{\boldsymbol{x}}}\kappa_{N_\ell}\right) \tag{4.4.2}$$

The output $F^{(\ell)}$ is not a function defined on $\mathcal{M}$, but a *section*, in general non smooth, of the *function bundle* $\mathbb{L}^2\left(SO(\mathcal{M}), \mathbb{R}^{N_\ell}\right) = \sqcup_{\boldsymbol{x}} L^2(SO(\boldsymbol{x}), \mathbb{R}^{N_\ell})$.
**Group correlation layer.** if $F$ is a function $SO(\boldsymbol{x}) \to \mathbb{R}^M$, we define $F \star_\gamma K$ as the *classical* group correlation

$$F \star_\gamma K(S) = \sum_{i=1}^{M} \int_{SO(\boldsymbol{x})} F_i(U) K_i(P_\gamma^{-1}S^{-1}UP_\gamma)\, dU. \tag{4.4.3}$$

This construction provides a new family of functions $\bar{F}_{\boldsymbol{x}} : SO(\boldsymbol{x}) \to \mathbb{R}$. Differing from [5], translations are in general not defined in $\mathcal{M}$ and rotations are only local. If $F_\gamma = (f_i\tilde{\star}_\gamma\kappa_i)_{i=1}^M$ and $F_\delta = (f_i\tilde{\star}_\delta\kappa_i)_{i=1}^M$ then it can be easily shown using the bi-invariance of the Haar measure on $SO(n)$ that $\varphi(F_\delta) \star_\delta K(S) = \varphi(F_\gamma) \star_\gamma K(SR)$ where $R$ depends only on paths $\gamma$ and $\delta$, and $\varphi$ is any real function (typically a rectified linear unit (ReLU)). With input $F^{(\ell-1)} \in \mathbb{L}^2\left(SO(\mathcal{M}), \mathbb{R}^{N_{\ell-1}}\right)$ with $N_{\ell-1}$ channels at layer $\ell - 1$ and

$\boldsymbol{x}_0$-rotation kernels $\boldsymbol{K}^{(\ell)} = \left(K_1^{(\ell)}, \ldots, K_{N_\ell}^{(\ell)}\right)$, each with $N_{\ell-1}$ channels, one obtains $F^{(\ell)} \in \mathbb{L}^2(SO(\mathcal{M}), \mathbb{R}^{N_\ell})$ as

$$F_{\boldsymbol{x}}^{(\ell)} = \left(F^{(\ell-1)} \star_{\gamma_{\boldsymbol{x}}} K_1^{(\ell)}, \ldots, F^{(\ell-1)} \star_{\gamma_{\boldsymbol{x}}} K_{N_\ell}^{(\ell)}\right) \tag{4.4.4}$$

**Projection layer.** A family $F^{(\ell-1)} \in \mathbb{L}^2\left(SO(\mathcal{M}), \mathbb{R}^{N_{(\ell-1)}}\right)$ is projected to a function $f : \mathcal{M} \to \mathbb{R}^{N_{(\ell-1)}}$ as

$$f_i^{(\ell)}(x) = \max_{S \in SO(\boldsymbol{x})} F_{i\,\boldsymbol{x}}^{(\ell-1)}(S), \quad i = 1 \ldots N_{\ell-1} \tag{4.4.5}$$

This removes the path dependency thanks to the change of path property which was described above. See Figure 4.1a for illustration.

Biases are added per kernel. Nonlinear transformations of ReLU type are applied after each of these layers. Note that without them, a lifting followed by group correlation would actually factor in a new lifting transformation.

### 4.4.2 Discretisation and implementation in the case $\mathcal{M} = \mathbb{S}^2$

In this work, the manifold of interest is $\mathbb{S}^2$. Spherical functions $f : \mathbb{S}^2 \to \mathbb{R}^N$ are typically given at a number of points and interpolated using a Watson kernel [26], which also serves as our choice. We use a very simple discretisation of $\mathbb{S}^2$ via the vertices of a regular icosahedron. Tangent kernels are defined over these vertices, sampled along with the rays of a polar coordinate system respecting the vertices of the icosahedron. This is illustrated in Figure 4.1b.

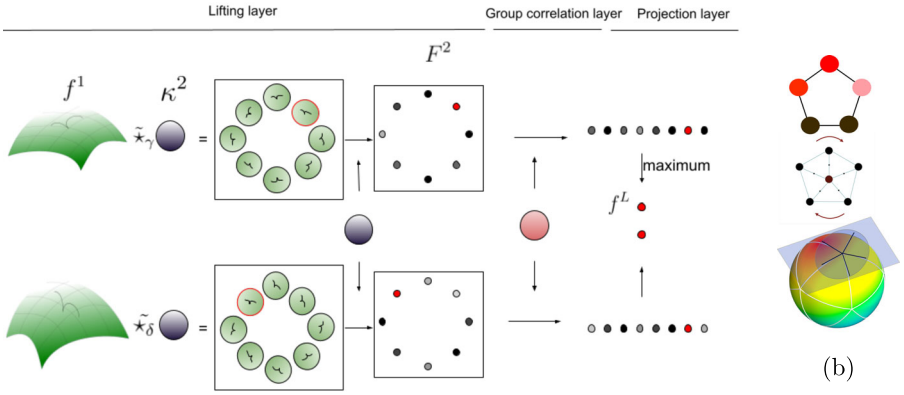
## 4.5 Experiments & Results

We evaluate our method on DWI data from the human connectome project [46]. We train a network using our framework on individual voxels containing signals on $\mathbb{S}^2$. Our goal is a voxel-wise classification of 4 regions of the brain - cerebrospinal fluid (CSF), subcortical, white matter, and grey matter regions.

We used the pre-processed DWI data [43] and normalised each DWI scan for the $b$-1000, $b$-2000, and $b$-3000 images respectively with the voxel-wise average of the $b_0$. The labels provided with the T1-image were transformed to the DWI using nearest neighbour interpolation (Figure 4.2). Since the 4 brain regions we are classifying have imbalanced numbers of voxels, we use Focal Loss [30] to counter the class imbalance of the dataset.

### 4.5.1 Experimental setup

After getting the responses from our proposed layers, we feed them into a small feedforward neural network to perform our classification task. To validate our




(a) (b)

Figure 4.1: In Figure 4.1a, the top row shows the lifting kernel $\kappa^{(2)}$ applied at a point on the manifold, resulting in an image $F^{(2)}$ defined on $SO(2)$ as in Equation (4.4.1). The function is first mapped onto the tangent space of the point of interest via the exponential map, and $\kappa^{(2)}$ is convolved with the mapped function to get $F^2$. In the figure we rotate the tangent space instead of the kernel as Equation (4.4.1) for convenience in illustration, but they are equivalent constructions. We get rotationally invariant responses from the projection layer. The bottom row shows the same process but with a different kernel parallel transport, illustrating that the responses of the convolutional layers are simply rotated. In Figure 4.1b, the bottom row shows $\mathbb{S}^2$ with a regular icosahedric tessellation and a tangent plane at one of the vertices and 5 sampled directions. The disk represents the kernel support. The middle row shows the actual discrete kernel used, with the $2\pi/5$ rotations and the top row is represents the lifted function on the discrete rotation group.

method, we compare the proposed framework with a baseline setup - feeding the smoothed signal values of each voxel directly into a feedforward neural network without our 3-layer convolution. In addition, we design different layer setups to evaluate both describability of our kernels and the nonlinearity of the task.

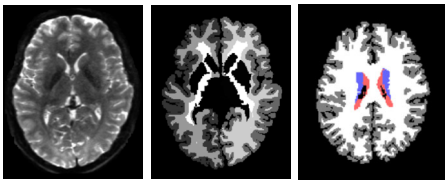
**Layer setup** We use 2 kinds of feedforward neural network structures in both the proposed method and the baseline experiments - single layer perceptron and multi layer perceptron. For the proposed method, connecting the responses from our 3-layer structure to a single layer perceptron without hidden layers tests the describability of our kernels while connecting the responses to a multi layer perceptron explores the full capacity of the method for the task. For baseline, feeding the smoothed signals directly to a single layer perceptron simply showcases the nonlinearity of the task, and the multi layer perceptron

| Experiment / Layer setup | Reduced model | Full model |
|---|---|---|
| Baseline | (90,4), DOF: 364 | (90,50,30,4), DOF: 6364 |
| Proposed method | (60,4), DOF: 286 | (60,30,4), DOF: 2056 |

Table 4.1: Illustration of the layer setups and degrees of freedom for our experiments.

setup, with nonlinearity added to the model, provides a generic comparison to the proposed method as a nonlinear model without dealing with the geometric encoding of the data. The network structures for both experiment setups are given in Table 4.1. For full models, each linear layer is followed by a ReLU activation and batch normalisation. For all models, the output at the last layer is followed by a softmax function to generate a probability map for the 4 classes.

In order to keep the model simple, we use the icosahedron structure as kernel locations with relatively low orientation resolution of the kernels - 5 rays per kernel, and 2 sample points per ray. The radius of the kernels should guarantee that the kernel coverage of 2 adjacent icosahedron vertices will overlap with each other, therefore we choose 0.6 as our radius. We use 1 kernel for the lifting layer, and 5 kernels for the group convolution layer to select 5 most essential structures of the signals.



(a) (b) (c)

Figure 4.2: (a)-(c): original diffusion data, the ground-truth segmentation, and the processed ground-truth that we are going to learn from. The label colors for CSF, subcortical, white matter and grey matter are red, blue, white and grey respectively. The figures are only for illustrations of the data, they are not necessarily from the same slice of the same scan.

### 4.5.2 Results

We use **1** scan for training, **1** scan for validation, and **50** scans for testing, all of which are with single-shell setup. We have observed that the overall validation loss and accuracy for all experiments converge after a few epochs. However, for the proposed method, for inner-class accuracies of difficult minority classes such as the subcortical region, the accuracy rises gradually towards convergence while not affecting the overall accuracy at all. Therefore, we use the convergence of subcortical region classification accuracy as a stopping criterion, which occurred after around 30 epochs. We train each network presented in Table 4.1 for 25 epochs with batch size 100 on an Ubuntu 20.04.2 LTS machine with an Intel Xeon(R) Silver 4210 CPU @ 2.20GHz $\times$ 40 processor and a GEFORCE RTX 3090 graphics card. Our framework is implemented in Python 3.6 and Pytorch 1.7.

For the proposed method, it takes around 4 min for an epoch and 1261MiB GPU memories for both experiment setups. For baseline experiments, it takes around 2 min and 3 min for an epoch, and 1261Mib and 1279Mib for the reduced and full model experiments respectively. We use $\kappa = 10$ for the interpolation using Watson kernel, and $\gamma = 2, \alpha = (0.15, 0.15, 0.35, 0.35)$ for focal loss, where the $\alpha$ weights correspond to CSF, subcortical regions, white matter, and grey matter respectively. Overall and inner-class accuracies and Dice scores are shown in Table 4.2.

| Results (Proposed/Baseline) / Layer setup | Reduced model | Full model |
|---|---|---|
| b=1000 | | |
| Overall accuracy | 0.78/0.598 | **0.798**/0.533 |
| CSF accuracy, Dice | 0.782/0.717, 0.782/0.768 | 0.769/**0.827**, **0.783**/0.728 |
| Subcortical accuracy, Dice | 0.18/0.014, 0.21/0.026 | 0.353/**0.689**, **0.348**/0.171 |
| White matter accuracy, Dice | 0.706/0.639, 0.777/0.601 | **0.767**/0.632, **0.805**/0.709 |
| Grey matter accuracy, Dice | **0.914**/0.628, 0.833/0.627 | 0.877/0.419, **0.844**/0.573 |
| b=2000 | | |
| Overall accuracy | **0.791**/0.71 | 0.779/0.562 |
| CSF accuracy, Dice | 0.706/0.68, **0.725**/0.709 | 0.676/**0.767**, 0.699/0.634 |
| Subcortical accuracy, Dice | 0.027/0.034, 0.048/0.052 | 0.433/**0.592**, **0.353**/0.163 |
| White matter accuracy, Dice | 0.778/0.628, **0.802**/0.684 | 0.734/**0.779**, 0.791/0.796 |
| Grey matter accuracy, Dice | **0.895**/0.861, 0.83/0.773 | 0.862/0.364, **0.831**/0.518 |
| b=3000 | | |
| Overall accuracy | **0.78**/0.775 | 0.774/0.698 |
| CSF accuracy, Dice | **0.627**/0.46, **0.62**/0.554 | 0.314/0.262, 0.414/0.301 |
| Subcortical accuracy, Dice | 0.271/0.002, 0.28/0.004 | 0.363/**0.407**, **0.33**/0.21 |
| White matter accuracy, Dice | 0.727/0.779, 0.791/0.779 | 0.717/**0.822**, 0.787/**0.805** |
| Grey matter accuracy, Dice | **0.891**/0.873, **0.831**/0.822 | 0.887/0.638, 0.827/0.73 |

Table 4.2: Results of both baseline and proposed method.

Firstly, across different $b$ values, we observe that with increased $b$, over all experiments, it becomes harder to recognise CSF. Secondly, across different experiment setups, we see from the single layer perceptron baseline experiment that the model performance improves with increasing $b$, yet the most difficult region to recognise - subcortical - was almost ignored by the model for all $b$. Higher $b$ provides more distinguishable signals for the majority classes, which contributes to the overall accuracy. Counterintuitively, the addition of nonlinearity to the baseline experiment - using multi layer perceptron - even worsens the performance. Adding nonlinearity does make the recognition of subcortical region more robust, but it is at a high cost of the misclassification of other classes, which is also why the Dice score for subcortical is still low while the accuracy is high for this experiment across all $b$ values (see Figure 4.3b). On the other hand, our proposed method shows robust performance generalising to the test set with far fewer degrees of freedom, and is stable across different $b$. Additionally, the full model of the proposed method - connecting our convolution layers to a multi layer perceptron - does a better job in recognising the

subcortical region than its reduced model counterpart without causing much misclassification of other classes as in the baseline experiments. This shows that recognising a difficult class requires geometric structure as well as higher degrees of freedom of the model. See Figure 4.3 for distributions of accuracies and Dice scores of the 4 classes across 50 test scans. We show statistics in Figure 4.3 for $b$-1000 scans, which are the most common single-shell data.

Another fact that is worth mentioning is that for $b$-3000, the validation accuracy for CSF fluctuates drastically for all experiments except for the reduced baseline model. This, in our opinion, is due to the fact that with much higher noise in the data, it becomes harder to recognise the diffusion in CSF in general, and stepping into a local minimum in the nonlinear models that disregards CSF will not cost much the overall loss since it is a minority class.

Moreover, we have also trained both the proposed method and the baseline with 10 scans for $b$-1000 to test how much the size of the dataset is influencing the results. It has mildly improved the classification accuracies of our method to around 0.8 and 0.81 for the reduced and full model respectively but shows no significant improvement for the baseline experiments. Therefore, we can conclude that our method can capture the most significant features with a very limited amount of data while the standard neural networks suffer from capturing geometric features even when the dataset is significantly increased.

See Figure 4.4 for examples of predictions from both proposed and baseline models, with $b$-1000.

## 4.6 Discussion and Conclusion

The proposed method is a simple extension of CNN to Riemannian Manifolds which learns rotationally invariant features. The Bundle G-CNN capability has been demonstrated on a simple non-flat manifold, $\mathbb{S}^2$, and has been used to build a voxel-wise classification of DWI data to recognise 4 brain regions, with an accuracy of 79.8% for single-shell data with the most common parameter setup $b$-1000. With a single-shell setup, our method, while taking the subcortical region into account, compares well with existing methods that have multi-shell input [51, 12], which do not classify the subcortical region. We also achieved similar or better results compared to image registration based methods [27]. Our method allows us to learn very general features from merely a single-shell scan, and the results show very robust generalisation across 50 scans in the test set. This work has promising applications in understanding patterns of pathology, structure, and connectivity. It is also desirable in the future to test our model trained with the HCP dataset on scans with a different number of diffusion gradients. We expect improvements by adding spatial correlations through a classical convolutional layer, and the correlation of our model to fractional anisotropy (FA) and NODDI is worth investigating as well. Additionally, we have so far only tested it on $\mathbb{S}^2$, however, an extension

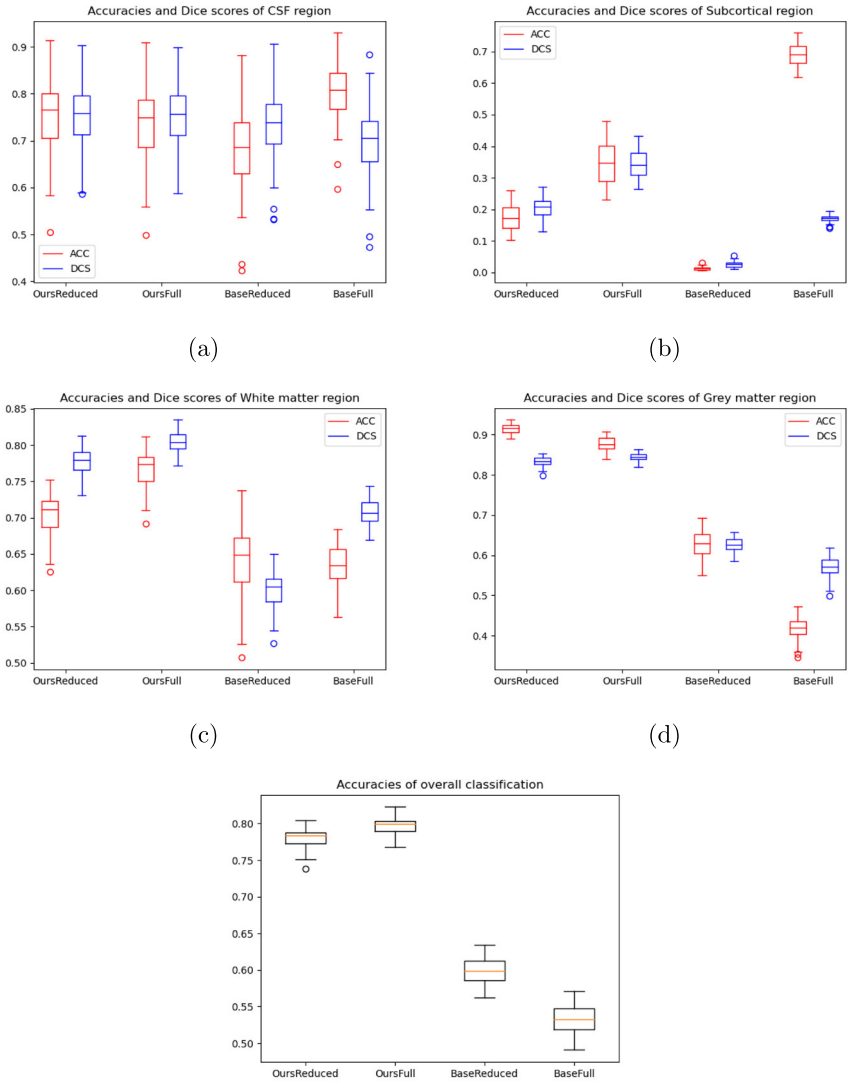
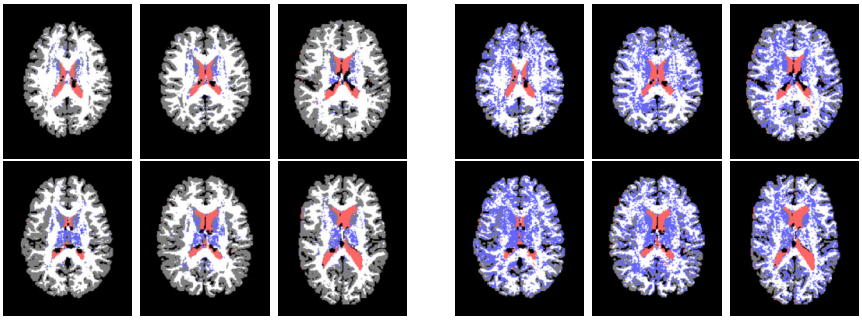



Figure 4.3: Statistics of accuracy and Dice score across individual scans of the 4 regions in the test set for $b$-1000.



(a) Prediction from proposed method with full model.

(b) Prediction from baseline experiment with full model.

Figure 4.4: Examples of predictions of the 4 regions in the test set. Predictions in Figure 4.4a are from the full model of the proposed method, while predictions of the same slices in Figure 4.4b are from the full model of the baseline experiment. The label colors for CSF, subcortical, white matter and grey matter are red, blue, white and grey respectively.

to other surfaces appears feasible, though the choice of a discrete representation is important. An extension to dimension 3 will require efficient $SO(3)$ convolutions, using, for instance, spectral theory for compact Lie groups.

# Chapter 5

# Bundle Geodesic Convolutional Neural Network for DWI Segmentation

Renfei Liu, François Lauze, Kenny Erleben, Rune W. Berg, Sune Darkner

## 5.1 Abstract

**Purpose** Applying machine learning techniques to Magnetic Resonance Diffusion-Weighted Imaging (DWI) data is challenging due to the size of individual data samples and the lack of labeled data. It possible though to learn general patterns from a very limited amount of training data if we take advantage of the geometry of the DWI data. Therefore, we present a tissue classifier based on a Riemannian Deep Learning framework for single-shell DWI data.
**Approach** The framework consists of three layers: a lifting layer that locally represents and convolves data on tangent spaces to produce a family of functions defined on the rotation groups of the tangent spaces, i.e., a (not necessarily continuous) function on a bundle of rotational functions on the manifold; a group convolution layer that convolves this function with rotation kernels to produce a new family of local functions over each of the rotation groups; and a projection layer using maximization to collapse this local data to form new manifold based functions.
**Results** Experiments show that our method achieves the performance of the same level as state-of-the-art while using way fewer parameters in the model (less than 10%). Meanwhile, we conducted a model sensitivity analysis for our method. We ran experiments using a proportion (69.2%, 53.3%, and 29.4%) of the original training set and analyzed how much data the model needs for

the task. Results show that this does reduce the overall classification accuracy mildly, but it also boosts the accuracy for minority classes.
**Conclusions** This work extended Convolutional Neural Networks (CNN) to Riemannian manifolds, and it shows the potential in understanding structural patterns in the brain, as well as in aiding manual data annotation.

## 5.2 Introduction

Studies for Magnetic Resonance Diffusion-Weighted Imaging (DWI) data have small sample sizes in general due to the lack of manually annotated data. For machine learning techniques, this poses a major challenge. However, learning general patterns from a limited amount of data while producing promising results is possible by conducting a special Geodesic Convolutional Neural Network (G-CNN) architecture that takes advantage of the geometry of the data. In general, to define convolution, the underlying space must have a group structure or be a homogeneous space of a group. This is not the case for most curved spaces. But even when it is, like the spheres on which local DWI signals are defined, this often imposes a certain type of filter. Instead, we present a framework that focuses on building a neural network (NN) for data on Riemannian manifolds with some simple form of orientation invariance, and we take DWI as the main application. There are a series of proposals trying to generalize a $\mathbb{R}^2$ convolutional neural network to curved spaces, yet in our case, rotational invariance is a desirable property we want in the design and our goal is to be able to understand spherical patterns up to rotations. We propose a general architecture for extracting and filtering local orientation information of data defined on a manifold that allows us to learn similar orientation structures which can appear at different locations on the manifold. Reasonable manifolds have local orientation structures – rotations on tangent spaces. Our architecture lifts data to these structures and performs local filtering on them, after which it collapses them back to obtain filtered features on the manifold. This provides both rotational invariance and flexibility in design, without having to resort to complex embeddings in Euclidean spaces. We provide an explicit construction of the architecture for DWI data and show very promising results for this case including learning and generalizing patterns from only one scan. This work is an extension of our previous publication [31].

## 5.3 Related work

The importance of the extraction of rotationally invariant features beyond Fractional Anisotropy [4] has been recognized in series of DWI works. [8] developed invariant polynomials of spherical harmonic (SH) expansion coefficients, and discussed their application in population studies. [38] proposed

a related construction using eigenvalue decomposition of SH operators. [35] and [52] argued their usefulness for understanding microstructures in relation to DWI.

There is though a vast growth in literature on Deep Learning (DL) for non-flat data or more complex group actions than just translations. [32] proposed a NN on surfaces that extracts local rotationally invariant features. A non-rotationally invariant modification was proposed in [7]. On the other hand, convolution generalises to more group actions than just translation, and this has led to group-convolution neural networks for structures where these operations are supported, especially Lie groups themselves and their homogeneous spaces [22, 16, 47, 50, 28, 5, 1]. Global equivariance is often sought but proved complicated or even elusive in many cases when the underlying geometry is non-trivial [44]. An elementary construction on a general manifold is proposed in [37] via a fixed choice of geodesic paths used to transport filters between points on the manifold, ignoring the effects of path dependency (holonomy). Removing this dependency can be obtained by summarising local responses over local orientations, this is what is done in [32]. To explicitly deal with holonomy, [42] proposed a convolution construction on manifolds based on stochastic processes via the frame bundle, but it is at this point still very theoretical.

A few works have applied DL to DWI. [24] built multi-layer perceptrons in $q$-space for kurtosis and NODDI mappings. Because of the spherical structure of the DWI data and the homogeneous structure of the sphere, [9] proposed an rotation equivariant construction inspired by [15] for disease classification. [34] propose a sixth-D, 3D space and $q$-space NNs with roto-translation / rotation equivalence properties.

In this work, we are interested in rotationally invariant features, so we take a path closer to [37, 32], but we add an extra local group convolution layer before summarising the data and eliminating path dependency. The proposed construction thus applies to oriented Riemannian manifolds, and no other structure (e.g. homogeneous or symmetric space) is used.

## 5.4 Method

All along this section our reference on Riemannian Geometry can be found in the textbook *Riemannian Geometry* [19]. CNNs are generally described and implemented in terms of correlation rather than convolution, and we follow this convention as well in this section. Bekkers et al. [5] used the fact that $SE(2)$ acts on $\mathbb{R}^2$ to lift 2D (vector-values) images to $\mathbb{R}^2 \times \mathbb{S}^1$ via *correlation kernels*. This is not in general the case when $\mathbb{R}^2$ is replaced by a Riemannian manifold, where there is no obvious way to define these operations. One can however overcome this situation via a somewhat more complex construction. Therefore, we assume in the sequel that we are given a complete orientable

Riemannian manifold of dimension $n$, this will be the sphere $\mathbb{S}^2$ in our case. We assume that the *injectivity radius* $i(\mathcal{M})$ of $\mathcal{M}$ is strictly positive. As usual, the tangent space at a point $\mathbf{x} \in \mathcal{M}$ is $T_{\mathbf{x}}\mathcal{M}$. An *image* is a function $f = (f_1 \ldots, f_{N_c}) \in L^2(\mathcal{M}, \mathbb{R}^{N_c})$, where $N_c$ is the number of channels.

Operations will be performed via lifting the function to tangent spaces and kernels are defined on tangent spaces. The exponential map $\mathrm{Exp}_{\mathbf{x}} : T_{\mathbf{x}}\mathcal{M} \to \mathcal{M}$ allows us to lift $f$ to $T_{\mathbf{x}}\mathcal{M}$ by setting $f_{\mathbf{x}} = (f_{i,\mathbf{x}})_i \equiv f \circ \mathrm{Exp}_{\mathbf{x}}$.

### 5.4.1 Layer definitions

**Lifting Layer.** We first define transportable filters on tangent spaces to replace CNN's kernels. These filters will also be called kernels. To start with, a "pointed kernel" will be a function $\boldsymbol{k} = (k_1, \ldots, k_{N_c}) \in L^2(T_{\mathbf{x}_0}\mathcal{M}, \mathbb{R}^{N_c})$, at a "base point $\mathbf{x}_0$". We assume that $\mathrm{supp}(\boldsymbol{k}) \in B_{\mathbf{x}_0}(0, r)$, $0 < r \leq i(\mathcal{M})$, the ball of center 0 and radius $r$ in $T_{\mathbf{x}_0}\mathcal{M}$. A piece-wise smooth path $\gamma : [0, 1] \to \mathcal{M}$, joining $\mathbf{x}_0$ to $\mathbf{x}$ defines, via the Levi-Civita connection of $\mathcal{M}$, *a parallel transport* $P_\gamma : T_{\mathbf{x}_0}\mathcal{M} \to T_{\mathbf{x}}\mathcal{M}$, and this is an isometry. We set $\boldsymbol{k}_\gamma \equiv \boldsymbol{k} \circ P_\gamma^{-1}$. In general, another smooth path $\delta : [0, 1] \to \mathcal{M}$ joining $\mathbf{x}_0$ and $\mathbf{x}$ defines another parallel transport $P_\delta : T_{\mathbf{x}_0}\mathcal{M} \to T_{\mathbf{x}}\mathcal{M}$ and $P_\gamma \circ P_\delta^{-1}$ is a *rotation* $R$ of $T_{\mathbf{x}}\mathcal{M}$, i.e., an element of $SO(T_{\mathbf{x}}\mathcal{M})$. It follows that $\boldsymbol{k}_\delta = \boldsymbol{k}_\gamma \circ R$. The $\gamma$-*lift* of $f$ by $\boldsymbol{k}$ is the the function

$$(\boldsymbol{k} \star_\gamma f)(S) = \sum_{i=1}^{N_c} \int_{T_{\mathbf{x}}\mathcal{M}} \kappa_{i,\gamma}(S^{-1}v) f_{i,\mathbf{x}}(v)\, dv, \quad S \in SO(T_{\mathbf{x}}\mathcal{M}). \tag{5.4.1}$$

Note that because $\mathrm{supp}(\boldsymbol{k}) \in B_{\mathbf{x}_0}(0, r)$, this integral is defined on $B_{\mathbf{x}}(0, r)$ and $Exp_{\mathbf{x}}$ is a diffeormorphism from this domain to the geodesic ball $B(\mathbf{x}, r) \subset \mathcal{M}$.

Now we choose, for each $\mathbf{x}$ in $\mathcal{M}$, a smooth path $\gamma_{\mathbf{x}}$ that joins $\mathbf{x}_0$ and $\mathbf{x}$. As $\mathcal{M}$ is complete, we can, for instance, choose a family $\Gamma = (\gamma_{\mathbf{x}})_{\mathbf{x}}$ of minimizing geodesics. The mapping

$$f \mapsto F = (\boldsymbol{k} \star_{\gamma_{\mathbf{x}}} f)_{\gamma_{\mathbf{x}} \in \Gamma}, \quad F(\mathbf{x}) : R \in SO(T_{\mathbf{x}}\mathcal{M}) \mapsto (\boldsymbol{k} \star_{\gamma_{\mathbf{x}}} f)(R) \in \mathbb{R} \tag{5.4.2}$$

lifts a $\mathcal{M}$-image to the *bundle of rotations* of $\mathcal{M}$ (we refer to Gallier et al. [21] chap. 9 for a definition of bundles in differential geometry), denoted by $\mathcal{SO}(T\mathcal{M})$ in the sequel ($\mathcal{SO}(T\mathcal{M})_{\mathbf{x}} = SO(T_{\mathbf{x}}\mathcal{M})$) as in Figure 5.1b. This lifting depends on the choice of the base point $\mathbf{x}_0$ and the choice of paths from $\mathbf{x}_0$ to any point $\mathbf{x}$ of $\mathcal{M}$. The *lifting layer* at level $(\ell - 1)$ takes a function $f : \mathcal{M} \to \mathbb{R}^{N_{\ell-1}}$ and uses $N_\ell$ kernels $(k^{1(\ell)}, \ldots, k^{N_\ell(\ell)})$ to produce

$$F^{(\ell)} = \left( k^{1(\ell)} \star_{\gamma_{\mathbf{x}}} f^{(\ell-1)}, \ldots, k^{N_\ell(\ell)} \star_{\gamma_{\mathbf{x}}} f^{(\ell-1)} \right)_{\gamma_{\mathbf{x}}}. \tag{5.4.3}$$

**Group Correlation Layer.** The object $F$ defined in (5.4.2) is a function on the total space of the bundle ($\mathcal{SO}(T\mathcal{M})$) (Gallier et al. [21] chap. 9), supposed square-integrable ($F \in \mathcal{L}^2(\mathcal{SO}(T\mathcal{M}))$).

The situation is more complex than the one described in Bekkers et al. [5] , as there is actually no reason that one can find a "continuous family" of paths $\mathbf{x}_0 \rightsquigarrow \mathbf{x}$, $\forall \mathbf{x} \in \mathcal{M}$. An important example to us: if $\mathcal{M}$ is the sphere $\mathbb{S}^2$, one can take $\gamma_{\mathbf{x}}$ to be a minimizing geodesic between $\mathbf{x}_0$ and $\mathbf{s}$. It is unique, except when $\mathbf{x} = -\mathbf{x}_0$, where there are infinitely many of them.

Let $K$ be an element of $L^2(T_{\mathbf{x}_0}\mathcal{M})$. The parallel transport of $K$ along the path $\gamma$ is $K_\gamma(R) = K(P_\gamma^{-1} R P_\gamma)$, as $P_\gamma^{-1} R P_\gamma \in SO(T_{\mathbf{x}_0}\mathcal{M})$. The correlation $F(\mathbf{x}) \star K_{\gamma_{\mathbf{x}}}$ is the group-theoretic one:

$$F(\mathbf{x}) \star K_{\gamma_{\mathbf{x}}}(S) = \int_{SO(T_{\mathbf{x}}\mathcal{M})} F(\mathbf{x})(R) K_{\gamma_{\mathbf{x}}}(S^{-1}R) dR \tag{5.4.4}$$

with $dR$ the bi-invariant Haar measure on $SO(T_{\mathbf{x}}\mathcal{M})$. In general, we consider objects that are a bit more complicated. Instead of $F$ being a section of $\mathcal{L}^2(\mathcal{SO}(T\mathcal{M}))$, it is taken as a section of $\mathcal{L}^2(\mathcal{SO}(T\mathcal{M}))^{N_l}$, meaning we have $N_l$ channels, $F(\mathbf{x}) = (F(\mathbf{x})_1, \ldots, F(\mathbf{x})_{N_l}) \in L^2\left(SO(T_{\mathbf{x}}\mathcal{M}), \mathbb{R}^{N_l}\right)$, and $K$ also has $N_l$ channels, $K = (K_1, \ldots, K_{N_l}) \in L^2(SO(T_{\mathbf{x}_0}), \mathbb{R}^{N_l})$ and we replace (5.4.4) by

$$F(\mathbf{x}) \star K_{\gamma_{\mathbf{x}}}(S) = \sum_{i=1}^{N_l} \int_{SO(T_{\mathbf{x}}\mathcal{M})} F(\mathbf{x})_i(R) K_{i,\gamma_{\mathbf{x}}}(S^{-1}R) dR \tag{5.4.5}$$

The group correlation layer at level $\ell$ takes a section $F$ of $\mathcal{L}^2(\mathcal{SO}(T\mathcal{M}))^{N_l}$, and uses $N_{\ell+1}$ kernels $(K_1^{(\ell+1)}, \ldots K_{N_{\ell+1}}^{(\ell+1)})$ to produce

$$F^{(\ell+1)} = \left(F^{(\ell)}(\mathbf{x}) \star K_{1,\gamma_x}^{(\ell+1),}, \ldots, F^{(\ell)}(\mathbf{x}) \star K_{N_{\ell+1},\gamma_x}^{(\ell+1)}\right)_{\mathbf{x}\in\mathcal{M}}$$

**Projection Layer.** The base point and path dependency in the lifting and group correlation layer definitions appear problematic. We can, however, re-project the results from these layers to standard functions on $\mathcal{M}$, *eliminating* this dependency. The only condition is that the same family of paths is used both in the lifting and group correlation layers to parallel transport the kernels.

Indeed, from what precedes, two $\gamma$- and $\delta$-lifts, though in general distinct, obey the simple relation

$$(k \star_\gamma f)(S) = (k \star_\delta f)(SR), \quad R = P_\gamma \circ P_\delta^{-1}. \tag{5.4.6}$$

A direct computation shows that $K_\delta(S) = K_\gamma(R^{-1}SR)$:

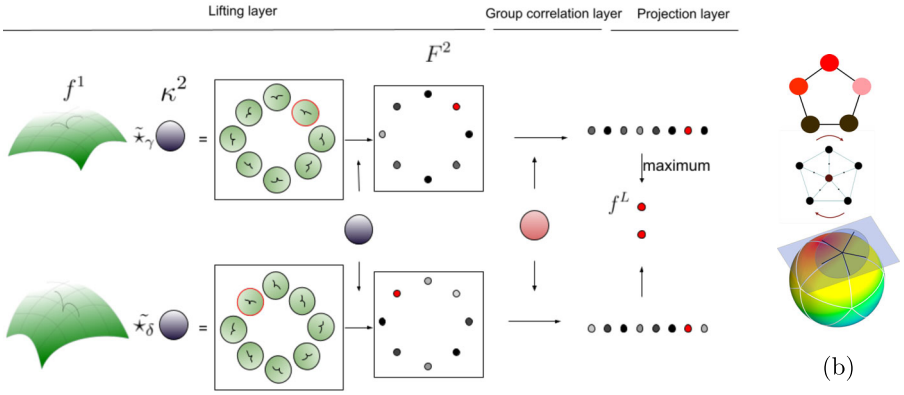
$$\begin{aligned}
((k \star_\delta f) \star K_\delta)(S) &= \int_{SO(T_{\mathbf{x}}\mathcal{M})} (k \star_\delta f)(T) K_\delta(S^{-1}T)\, dT \\
&= \int_{SO(T_{\mathbf{x}}\mathcal{M})} (k \star_\gamma f)(TR) K_\gamma(R^{-1}S^{-1}TR)\, dT \\
&= \int_{SO(T_{\mathbf{x}}\mathcal{M})} (k \star_\gamma f)(U) K_\gamma(R^{-1}S^{-1}U)\, dU, \quad U \leftarrow TR \\
&= ((k \star_\gamma f) \star K_\gamma)(SR)
\end{aligned}$$

where we used the fact that the normalized Haar measure on $SO(T_{\mathbf{x}}\mathcal{M})$ is bi-invariant, thus in particular right-invariant.

Thus the following projection layer is well-defined and removes the base point and path dependency:

$$f^{\ell+2}(\mathbf{x}) = \max_{R \in SO(T_{\mathbf{x}}\mathcal{M})} F^{(\ell+1)}(\mathbf{x}, R). \tag{5.4.7}$$

Biases are added per kernel. Nonlinear transformations of ReLU type are applied after each of these layers. Note that without them, a lifting followed by group correlation would actually factor in a new lifting transformation.




(a) (b)

Figure 5.1: In Figure 5.1a, the top row shows the lifting kernel $\kappa^{(2)}$ applied at a point on the manifold, resulting in an image $F^{(2)}$ defined on $SO(2)$ as in Equation (5.4.2). The function is first mapped onto the tangent space of the point of interest via the exponential map, and $\kappa^{(2)}$ is convolved with the mapped function to get $F^2$. Group correlation is then performed on the resulting image, followed by the projection layer, from which we get rotationally invariant responses. The bottom row shows the same process but with a different kernel parallel transport, illustrating that the responses of the convolutional layers are simply rotated. In Figure 5.1b, the bottom row shows $\mathbb{S}^2$ with a regular icosahedric tessellation and a tangent plane at one of the vertices and 5 sampled directions. The disk represents the kernel support. The middle row shows the actual discrete kernel used, with the $2\pi/5$ rotations and the top row is represents the lifted function on the discrete rotation group.

### 5.4.2 Discretization and implementation in the case $\mathcal{M} = \mathbb{S}^2$

In this work, the manifold of interest is $\mathbb{S}^2$. Spherical functions $f : \mathbb{S}^2 \to \mathbb{R}^N$ are typically given at a number of points and interpolated using a Watson kernel [26], which also serves as our choice. We use a very simple discretization

of $\mathbb{S}^2$ via the vertices of a regular icosahedron. Tangent kernels are defined over these vertices, sampled along with the rays of a polar coordinate system respecting the vertices of the icosahedron. This is illustrated in Figure 5.1b.

## 5.5 Experiments and Results

We evaluate our method on 3 datasets: a DWI scan conducted on a spinal cord that had been dissected out post mortem from a deceased human female, a synthetic dataset that we generated, and the DWI brain scan dataset from the human connectome project [46].

#### 5.5.0.1 Experimental setup

After getting the responses from our proposed layers, we feed them into a small feedforward neural network - a single layer perceptron - to perform our classification task. To validate our method, we compare the proposed framework with 2 experimental setups: a) a baseline experiment that feeds the smoothed signal values of each voxel directly into a feedforward neural network without our 3-layer convolution; b) S2CNN [15] which performs convolution on spheres by transforming the signals onto the spectral domain. For all the experiments, we use the smallest model possible for both our method and S2CNN [15].

### 5.5.1 Spinal Scan

#### 5.5.1.1 Data Description

The study was conducted on a deceased individual who had bequeathed her body to science and education at the Department of Cellular and Molecular Medicine (ICMM) of the University of Copenhagen according to Danish legislation (Health Law No. 546, Section 188). The study was approved by the head of the Body Donation Program at ICMM. Part of the data used here has been published in a previous report [25]. Briefly, the spinal cord was dissected out from a 91-year old Caucasian female without known diseases post mortem within 24h after her death. The spinal cord was fixed by immersion into paraformaldehyde (4%), where it was kept for 2 weeks, after which it was transferred to and stored in phosphate-buffered saline until the MRI scanning was conducted. The spinal cord was placed in a plexiglas tube and immersed in fluorinert (FC-40, Sigma-Aldrich) to eliminate any background signal. The scanning was accomplished using a 9.4T preclinical system (BioSpec 94/30; Bruker Biospin, Ettlingen, Germany) equipped with a 1.5 T/m gradient coil. The scanning was done in 29 sections of length 1.6 cm, thus covering the whole length of the spinal cord of approximately 40 cm. Between each section scan,

the tissue was advanced 1.4 cm by a custom-built stepping motor system, resulting in a 0.2-cm section overlap. For each section, a T2-weighted 2D RARE structural scan was performed. Scan parameters were repetition time (TR) = 7 s, echo time (TE) = 30 ms, 20 averages, field of view $1.92 \cdot 1.92 \cdot 1.6\ cm^3$, and a matrix size of $384 \cdot 384 \cdot 80$, resulting in $50 \cdot 50\ \mu m^2$ in-plane resolution and a slice thickness of 200 $\mu m$, resulting in a voxel size of 500000 $\mu m^3$ . The scan time for the structural scan was 30h.

We take individual voxels containing signals defined on $\mathbb{S}^2$ as the input of the networks and achieve segmentation via voxel classification. Since the numbers of samples of white matter and grey matter are not balanced, we use Focal Loss[30] to counter the imbalance. We used 14 slices from the longest dimension to test and the rest of the scan to train.

**Architecture and Hyperparameters** For our method, we use the icosahedron structure as kernel locations, and a $lift - ReLU - conv - ReLU - projection - FC - softmax$ architecture for the network. We use $k = 1, 5, 2$ channels for $lift$, $conv$, and $FC$, 0.6 as kernel radius, and 5 rays, 2 samples per ray as kernel resolution. For S2CNN [15], we use the simple architecture they provided $S^2conv - ReLU - SO(3)conv - ReLU - FC - softmax$, bandwidth $b = 30, 10, 6$ and $k = 4, 8, 2$ channels. For baseline, we use $FC(80) - ReLU - FC(50) - ReLU - FC(30) - ReLU - FC(2)$ as a multi layer perceptron alternative. We use $\kappa = 10$ for the Watson kernel, 0.001 as learning rate, and trained each model for 20 epochs. We use $\gamma = 2$, and $\alpha = (0.25, 0.75)$ for white and grey matter respectively for the Focal Loss[30].

#### 5.5.1.2 Results

We can see from Table 5.1 that all methods perform quite well for this simple task. Showcase of prediction from our model and the ground-truth can be found in Figure 5.2. We observe that classifying white matter and grey matter is not a challenging task considering the baseline model works well for this task. This is because there is already a significant difference between white matter and grey matter in terms of the scales of the intensity values of the two tissues. However, our method and S2CNN[15] have a better balance between the accuracies of the two classes compared to the baseline, which shows the importance of geometric information for recognizing minority classes. To test the rotational invariance and the independence to scaling of the signals of our method, we experiment further on the synthetic dataset and the HCP dataset [46].

| Results / Experiment (#Param) | Our method (164) | Baseline (5802) | S2CNN (6270) |
|---|---|---|---|
| b=4000 | | | |
| Overall Acc | **0.902** | 0.897 | 0.887 |
| White matter Acc | 0.905 | **0.911** | 0.891 |
| Grey matter Acc | **0.883** | 0.833 | 0.872 |

Table 5.1: Results from the spinal scan. The numbers in the brackets are numbers of parameters for each model. We see that overall, all methods achieve similar performance, yet convolution involved methods - ours and S2CNN[15] - perform better in recognizing the minority class - grey matter.

Figure 5.2: Examples of ground-truth and predictions from the test data. From left to right are the same slices from the ground-truth, prediction from our method, prediction from S2CNN, and prediction from baseline.

### 5.5.2 Synthetic Dataset

#### 5.5.2.1 Dataset Generation

To validate the robustness of our method against rotations, we create and classify spherical functions that are defined on a sphere. We first uniformly sample 90 directions on a hemisphere, and spherical functions of different classes are defined in the same 90 directions. For each class, we sample 90 values from a Gaussian distribution as function values of the 90 directions. Thus the only difference among classes is the function values of the given 90 directions, and we sample the function values for each class from the same Gaussian distribution to keep the scales of the values identical. In addition, we rotate the sphere of each class and use these rotated spherical functions as elements of each class. Therefore, each class of the dataset contains just rotations of each spherical function. As explained above, we interpolate the function values at the icosahedron vertices using a Watson kernel[26]. For the baseline, we interpolate the function values at the same 90 directions that were sampled on the sphere using the same scheme.

We generate synthetic datasets of different numbers ($n \in 2, 4, 6$) of classes to test the robustness of the model, given different difficulties of the task. For each class, we generate 50 samples for the training set and 1000 samples for the test set.

**Architecture and Hyperparameters** As in the experiment above, we use a $lift - ReLU - conv - ReLU - projection - FC - softmax$ architecture for

the network. We use $k = 1, 5, n$ channels for $lift$, $conv$ and $FC$ layers, 0.6 as kernel radius, and 5 rays, 2 samples per ray as kernel resolution. For S2CNN [15], we use $S^2conv - ReLU - SO(3)conv - ReLU - FC - softmax$ as in the experiments above, bandwidth $b = 30, 10, 6$ and $k = 3, 6, n$ channels. For baseline, we use $FC(90) - ReLU - FC(50) - ReLU - FC(30) - ReLU - FC(n)$ as the multi layer perceptron layer structure. We use $\kappa = 5$ for the Watson kernel, 0.005 as learning rate, and trained each model for 200 epochs.

#### 5.5.2.2 Results

See Table 5.2 for comparison of results from different models. We can see that the baseline model is barely learning anything from the data, while our method and S2CNN [15] are capturing the differences from different classes in the data. Moreover, our method achieves more robust performance while having fewer degrees of freedom.

| # Classes / Experiment | Ours | Baseline | S2CNN |
|---|---|---|---|
| 2 | **1.0** (164) | 0.515 (6302) | 0.985 (3551) |
| 4 | **0.987** (286) | 0.256 (6364) | 0.966 (3565) |
| 6 | **0.984** (408) | 0.168 (6426) | 0.972 (3579) |

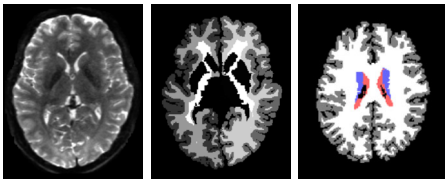
Table 5.2: Test accuracy for models evaluated on the generated datasets. Numbers in the brackets are the numbers of parameters for each model. The baseline model is producing prediction accuracies that are the same level as random guessing, while ours and S2CNN[15] can recognize the rotations of the same spherical functions quite accurately, and our method achieves higher accuracy using fewer parameters than S2CNN[15]

### 5.5.3 Human Connectome Brain Scans

As in the spinal data experiments, we train networks on individual voxels containing signals on $\mathbb{S}^2$. Our goal is a voxel-wise classification of 4 regions of the brain - cerebrospinal fluid (CSF), subcortical, white matter, and grey matter regions.

We used the pre-processed DWI data [43] and normalized each DWI scan for the $b$-1000, $b$-2000, and $b$-3000 images respectively with the voxel-wise average of the $b_0$.

The labels provided with the T1-image were transformed to the DWI using nearest neighbor interpolation (Figure 5.3). Since the 4 brain regions we are classifying have imbalanced numbers of voxels, we use Focal Loss [30] to counter the class imbalance of the dataset just as in the spine data experiments.



(a) (b) (c)

Figure 5.3: (a)-(c): original diffusion data, the ground-truth segmentation, and the processed ground-truth label image. The label colors for CSF, subcortical, white matter, and grey matter are red, blue, white, and grey respectively. The figures are only for illustrations of the data, they are not necessarily from the same scan.

**Architecture and Hyperparameters** As in experiments above, we use the icosahedron structure as locations for kernels for our method, and $lift - ReLU - conv - ReLU - projection - FC - softmax$ as network architecture with $k = 1, 5, 4$ channels, $r = 0.6$ as radius, and 5 rays with 2 samples per ray as kernel resolution. For S2CNN[15], we again use the same architecture provided by the authors $S^2conv - ReLU - SO(3)conv - ReLU - FC - softmax$, bandwidth $b = 30, 10, 6$ and $k = 3, 6, 4$ channels. For baseline, we again use $FC(90) - ReLU - FC(50) - ReLU - FC(30) - ReLU - FC(4)$ architecture. We use $\kappa = 10$ for the Watson kernel, and 0.001 as learning rate for all models. We use $\gamma = 2$ and $\alpha = (0.35, 0.35, 0.15, 0.15)$ for the Focal Loss[30] for the 4 regions respectively. Additionally, we have observed that the most difficult class to identify is the subcortical region, and both our method and S2CNN [15] learn to recognize it gradually. Therefore, we stop the training for all models when the subcortical region validation accuracy stops rising. Thus we train all models for 50 epochs.

#### 5.5.3.1 Results

We used **1** scan for training, **1** scan for validation, and **50** scans for testing. Comparison of experimental results of different methods can be found in Table 5.3. We can see that the baseline experiment does not generalize well compared to our method and S2CNN [15]. Across different $b$ values, we observe that with increased $b$, over all experiments, it becomes harder to recognize CSF. Higher $b$ does not reduce the accuracies for the majority classes for our method and S2CNN[15], thus the overall accuracies from these methods do not drop much with increased $b$. On the other hand, while comparing to S2CNN [15], we achieve very similar results yet our model has way lower degrees of freedom while achieving the same level of performance as we can see in Table 5.3.

#### 5.5.3.2 Model Sensitivity Analysis

We reduce the amount of training data for our method in order to test how sensitive our model is. As mentioned above, there is only 1 scan in the training

| Results / Experiment(DOF) | Our method (286) | Baseline (6364) | S2CNN (3565) |
|---|---|---|---|
| b=1000 | | | |
| Overall Acc | **0.791** ± 0.012 | 0.492 ± 0.015 | 0.784 ± 0.012 |
| CSF Acc, Dice | 0.785 ± 0.074,**0.747** ± 0.073 | **0.824** ± 0.06,0.577 ± 0.106 | 0.783 ± 0.075,0.744 ± 0.074 |
| Subcortical Acc, Dice | 0.201 ± 0.057,0.239 ± 0.052 | **0.495** ± 0.031,0.128 ± 0.011 | 0.299 ± 0.059,**0.276** ± 0.039 |
| White matter Acc, Dice | 0.778 ± 0.026,0.802 ± 0.014 | 0.67 ± 0.016,0.691 ± 0.012 | **0.816** ± 0.023,**0.81** ± 0.012 |
| Grey matter Acc, Dice | **0.872** ± 0.017,**0.835** ± 0.011 | 0.327 ± 0.026,0.483 ± 0.028 | 0.814 ± 0.02,0.827 ± 0.012 |
| b=2000 | | | |
| Overall Acc | 0.787 ± 0.012 | 0.452 ± 0.017 | **0.794** ± 0.011 |
| CSF Acc, Dice | 0.552 ± 0.075,0.612 ± 0.078 | **0.76** ± 0.07,0.605 ± 0.098 | 0.753 ± 0.079,**0.684** ± 0.088 |
| Subcortical Acc, Dice | 0.184 ± 0.045,**0.222** ± 0.042 | **0.694** ± 0.021,0.144 ± 0.008 | 0.123 ± 0.03,0.166 ± 0.034 |
| White matter Acc, Dice | 0.804 ± 0.032,0.806 ± 0.015 | 0.689 ± 0.022,0.748 ± 0.015 | **0.843** ± 0.025,**0.817** ± 0.012 |
| Grey matter Acc, Dice | **0.85** ± 0.023,0.827 ± 0.012 | 0.207 ± 0.027,0.339 ± 0.036 | 0.832 ± 0.021,**0.83** ± 0.011 |
| b=3000 | | | |
| Overall Acc | 0.786 ± 0.012 | 0.686 ± 0.016 | **0.788** ± 0.011 |
| CSF Acc, Dice | 0.203 ± 0.029,0.284 ± 0.04 | 0.188 ± 0.014,0.222 ± 0.028 | **0.303** ± 0.055,**0.341** ± 0.069 |
| Subcortical Acc, Dice | 0.216 ± 0.057,0.248 ± 0.05 | **0.358** ± 0.023,0.186 ± 0.015 | 0.228 ± 0.064,**0.256** ± 0.055 |
| White matter Acc, Dice | 0.767 ± 0.034,0.805 ± 0.018 | **0.83** ± 0.021,0.797 ± 0.011 | 0.783 ± 0.033,**0.812** ± 0.016 |
| Grey matter Acc, Dice | **0.888** ± 0.02,**0.832** ± 0.011 | 0.616 ± 0.037,0.714 ± 0.024 | 0.873 ± 0.021,0.831 ± 0.012 |

Table 5.3: Results from the HCP brain dataset. We can see that our method has the same level of performance as S2CNN[15], but uses way fewer parameters. The baseline model produces higher accuracy recognizing the subcortical region, but it is at a high cost of the accuracies of other classes.

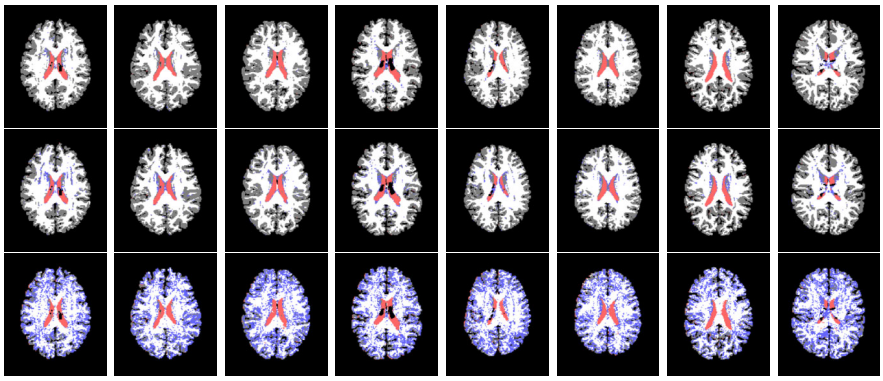


Figure 5.4: Examples of predictions of the 4 regions in the test set. From the top to bottom row are predicted results of the same slices from the proposed method, S2CNN[15], and baseline respectively. The label colors for CSF, subcortical, white matter and grey matter are red, blue, white, and grey respectively.

| Results / #Samples | 5000,30000,200000,200000 | 5000,30000,150000,150000 | 5000,30000,75000,75000 |
|---|---|---|---|
| b=1000 | | | |
| Overall accuracy | $0.784 \pm 0.013$ | $0.775 \pm 0.014$ | $0.718 \pm 0.024$ |
| CSF accuracy, Dice | $0.719 \pm 0.086$,$0.747 \pm 0.073$ | $0.776 \pm 0.076$,$0.752 \pm 0.073$ | $0.853 \pm 0.055$,$0.704 \pm 0.082$ |
| Subcortical accuracy, Dice | $0.271 \pm 0.072$,$0.279 \pm 0.051$ | $0.289 \pm 0.056$,$0.247 \pm 0.032$ | $0.643 \pm 0.061$,$0.293 \pm 0.025$ |
| White matter accuracy, Dice | $0.736 \pm 0.026$,$0.791 \pm 0.015$ | $0.746 \pm 0.029$,$0.789 \pm 0.016$ | $0.635 \pm 0.042$,$0.733 \pm 0.026$ |
| Grey matter accuracy, Dice | $0.887 \pm 0.017$,$0.832 \pm 0.012$ | $0.857 \pm 0.018$,$0.834 \pm 0.012$ | $0.794 \pm 0.028$,$0.819 \pm 0.017$ |
| b=2000 | | | |
| Overall accuracy | $0.777 \pm 0.014$ | $0.771 \pm 0.017$ | $0.729 \pm 0.017$ |
| CSF accuracy, Dice | $0.711 \pm 0.089$,$0.695 \pm 0.087$ | $0.744 \pm 0.079$,$0.682 \pm 0.088$ | $0.772 \pm 0.074$,$0.676 \pm 0.09$ |
| Subcortical accuracy, Dice | $0.24 \pm 0.039$,$0.235 \pm 0.028$ | $0.362 \pm 0.061$,$0.293 \pm 0.033$ | $0.462 \pm 0.043$,$0.24 \pm 0.018$ |
| White matter accuracy, Dice | $0.737 \pm 0.033$,$0.789 \pm 0.018$ | $0.735 \pm 0.037$,$0.789 \pm 0.02$ | $0.716 \pm 0.031$,$0.78 \pm 0.018$ |
| Grey matter accuracy, Dice | $0.876 \pm 0.019$,$0.83 \pm 0.012$ | $0.851 \pm 0.022$,$0.826 \pm 0.013$ | $0.769 \pm 0.031$,$0.802 \pm 0.015$ |
| b=3000 | | | |
| Overall accuracy | $0.785 \pm 0.011$ | $0.772 \pm 0.013$ | $0.662 \pm 0.023$ |
| CSF accuracy, Dice | $0.603 \pm 0.106$,$0.584 \pm 0.112$ | $0.525 \pm 0.082$,$0.528 \pm 0.102$ | $0.701 \pm 0.09$,$0.568 \pm 0.112$ |
| Subcortical accuracy, Dice | $0.288 \pm 0.068$,$0.277 \pm 0.047$ | $0.388 \pm 0.068$,$0.295 \pm 0.037$ | $0.63 \pm 0.048$,$0.222 \pm 0.018$ |
| White matter accuracy, Dice | $0.798 \pm 0.028$,$0.809 \pm 0.013$ | $0.761 \pm 0.028$,$0.801 \pm 0.014$ | $0.596 \pm 0.038$,$0.719 \pm 0.027$ |
| Grey matter accuracy, Dice | $0.836 \pm 0.026$,$0.829 \pm 0.012$ | $0.833 \pm 0.027$,$0.826 \pm 0.012$ | $0.722 \pm 0.042$,$0.78 \pm 0.02$ |

Table 5.4: Results of sensitivity analysis. The numbers in the first row are the numbers of samples in each experiment for CSF, subcortical, WM, and GM respectively. We see that while reducing the size of the training set, the overall accuracies decrease to some extent, but the accuracies of the subcortical region are higher since the class imbalance is lower.

set. For that scan, there are 7227 CSF voxels, 35648 subcortical voxels, 276191 white matter voxels, and 309496 grey matter voxels. Therefore, we reduce the number of samples from all classes by randomly sampling a fraction of voxels from each class and test how that impacts the performance of the model.

We see that reducing the number of samples in each class reduces the performance. On the other hand, it has also boosted the accuracy for the subcortical region, since that the class imbalance was also eased after the reduction. We can observe that the grey matter and white matter tissues are overly represented in a scan that even when we discard most of the voxels from these 2 classes in the training set, our test result remains a relatively high level of accuracy. This offers us an important application in automating DWI data annotation.

## 5.6 Discussion

This work shows how geometric information in DWI can be significantly useful in understanding general patterns in image analysis. In the future, we expect improvements in performance by adding spatial correlations through a classical convolutional layer, and the correlation of our model to fractional anisotropy (FA) and NODDI is worth investigating as well. Moreover, using scans in the HCP dataset[46] with a different number of diffusion gradients to test our model would also be desirable in later works. Additionally, we have so far only tested our construction of the network on $\mathbb{S}^2$, yet an extension to

other surfaces appears feasible, with a smart choice of a discrete representation. An extension to dimension 3 is worthwhile as well, which will require efficient $SO(3)$ convolutions, using, for instance, spectral theory for compact Lie groups.

## 5.7 Conclusion

We proposed a simple extension of CNN to Riemannian Manifolds that learns rotationally invariant features. Our method allows us to learn general patterns from very limited data while having much lower degrees of freedom than existing methods [15]. This is significant because we can now, in machine learning-based DWI analysis, reduce the size of individual data samples to a single voxel-level from a whole volumetric image-level, as well as reduce the training dataset to a single scan - or a fraction of a scan. For the HCP dataset[46] with a single-shell setup, our method, while taking the subcortical region into account, compares well with existing methods that have multi-shell input [51, 12], which do not classify the subcortical region. We also achieved similar or better results compared to image registration-based methods [27]. The results of this simple task show great potential of this method in understanding structural patterns in brains. Moreover, the results from the model sensitivity analysis show that our method has the potential in aiding manual data annotation. For example, a doctor only has to label a fraction of a scan and the rest can be automated by the model.

# Chapter 6

# Group Convolutional Neural Network for DWI Segmentation

Renfei Liu, François Lauze, Erik J. Bekkers, Kenny Erleben, Sune Darkner

## 6.1 abstract

We present a Group Convolutional Network for Segmentation of Diffusion Weighted Imaging data (DWI). The network incorporates group actions that are natural for this type of data, in the form of convolutions which provide equivariant transformations of the data. This knowledge provides a potentially important inductive bias and may alleviate the need for data augmentation strategies. We study its effect on the performances of the networks, by training them and validating them on DWI scans from the Human Connectome project. We show how this generally improves the performances of our segmentation, while limiting the number of parameters that must be learned.

## 6.2 Introduction

In this work, we propose a group convolutional neural network (G-CNN) for Diffusion Weighted Imaging (DWI) data. CNNs rely on assumed translational symmetries in data and have shown very robust performance in imaging tasks, especially medical imaging ones, and they are highly memory-efficient thanks to their weight-sharing property. When data offer more structure than translation, they can be used to build generalized CNNs. These Group and Geometric CNNs (GCNN) have been studied intensively and applied in many situations in the few past years, see e.g. [32, 16, 7, 5, 14]. This is especially the case for

the task at hand, classification and segmentation of DWI data. A DWI scan can be modeled as a function $f : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ providing at each $(x, v)$, with spatial position $x$ and direction $v$, a response ([45]). A rigid transformation of a sample (i.e. by the action of the group $SE(3)$), should be reflected, up to the limitations of acquisition protocol, in the signal. The space $\mathbb{R}^3 \times \mathbb{S}^2$ is a *homogeneous space* under the action of $SE(3)$: a point in $\mathbb{R}^3 \times \mathbb{S}^2$ can be transformed in any other point by a rigid transformation. This notion of homogeneous space is at the heart of the extension of CNNS to GCNNs [14, 6].

Our task at hand is the classification/segmentation of diffusion data. The inductive bias provided by the knowledge of these transformations may prove important for our task, especially when annotated data is limited. How to incorporate this knowledge? This is classically done by data augmentation, in the hope that the network will learn transformation-aware features during the training phase. Incorporating, on the other hand, information about group actions on the data has shown to boost performances of these networks [5]. To exploit the rigid motion action in the space of DWI - $\mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$, we propose an $SE(3)$-GCNN.

Most CNNs approaches for processing of DWI signals discard its specific structure. For instance, Golkov et al. [24] built multi-layer perceptrons in $q$-space for kurtosis and NODDI mappings. On the other hand, the importance of spherical equivariant or invariant structure has been acknowledged for some years now. The importance of the extraction of rotationally invariant features beyond Fractional Anisotropy [4] has been recognized in series of DWI works, for their usefulness in understanding microstructures, see for instance [38, 8, 35, 52]. In [40], a spherical U-net based on [20] was used for neurite orientation. Cohen et al. [15] lifted spherical functions to the 3D-rotation group $SO(3)$ an extension of Fourier transform on it to perform convolution. In [39], this idea was used for microstructure parameter estimation. In [9, 3], it was used for disease classification. In [34], a 6-D - 3D space and $q$-space - NNs with roto-translation was proposed.

Several authors [22, 16, 47, 50, 28, 5, 1, 44, 14, 2] further explored the group convolution path for Lie groups and their homogeneous spaces.

In the rest of this paper, we propose GCNNs with two types of group action, with equivariant layers for these actions and nonlinear ones. We show how incorporating these actions improve DWI segmentation performance compared to classical CNNs and CNNS with limited notion of symmetry [31], by evaluating them on scans from the Human Connectome Project (HCP) [46].

## 6.3 Method

The networks we present are built by extending standard CNNs to groups $G$ and homogeneous spaces $\mathcal{M}$ on which they act by extending convolution op-

erations to them. We do not follow, however, the common path of irreducible representations for implementing convolutions/correlations over $\mathbb{S}^2$ or $SO(3)$.

An action of a Lie group on a space $\mathcal{M}$ is, for our purpose a smooth mapping $G \times \mathcal{M} \to \mathcal{M}$, $(g, m) \to g.m$ such that for each $g$, $m \to g.m$ is a diffeomorphism of $\mathcal{M}$ and such that $g.(g'.m) = (gg').m$. The neutral element of $G$ acts as the identity. The orbit of $m \in \mathcal{M}$ is the set $G.m = \{g.m, g \in G\}$. The stabilizer $G_m$ of an element $m$ is the set of transformations that lets $m$ fixed, $G_m = \{g \in G, g.m = m\}$. It is a subgroup of $G$. $\mathcal{M}$ is a *$G$-homogeneous space* if it contains only one orbit. A point $m_0$ in the homogeneous space $\mathcal{M}$ provides an isomorphism $G/G_{m_0} \simeq \mathcal{M}$ from the quotient space $G/G_{m_0}$ and consists of the *left cosets* $gG_{m_0}$ of $G_{m_0}$. The inverse of the point $m$ by this map is the *coset* $gG_{m_0}$, with $g.m_0 = m$, also called the *fibre* above $m$. A group $G$ acting on a space $\mathcal{M}$ also acts on its functions on $\mathcal{M}$ by the *left translation* $(L_g f)(m) = f(g^{-1}m)$.

### 6.3.1 Standard convolution operations

Each group $G$ we consider is endowed with a left-invariant Haar measure. Each homogeneous space we consider is endowed with a $G$-invariant measure. Functions are assumed to be square-integrable for these measures. The layers $\mathcal{L}$ we define are all equivariant $\mathcal{L}(L_g f) = L_g(\mathcal{L}f)$, w.r.t. the regular representation $L_g$.

**Lifting layer.** A function $f : \mathcal{M} \to \mathbb{R}^N$ can be *lifted* to the group $G$ via a kernel $\kappa : \mathcal{M} \to \mathbb{R}^N$ by

$$\kappa * f(g) = \sum_{i=1}^{N} \int_{\mathcal{M}} f_i(m)\kappa_i(g^{-1}m)\, dm \tag{6.3.1}$$

**Group convolution layer.** A feature function $F : G \to \mathbb{R}^N$ can be transformed by a convolution kernel $K : G \to \mathbb{R}^N$ by

$$K * F(g) = \sum_{i=1}^{N} \int_{G} F_i(h)K_i(h^{-1}g)dh. \tag{6.3.2}$$

**Projection layer.** If needed, feature map $F : G \to \mathbb{R}^n$ can be projected to a function $f : \mathcal{M} \to \mathbb{R}^n$ by summarizing on the fibres

$$\overline{F}(m) = \max_{h \in G_{m_0}} F(gh), \quad \text{for any } g \text{ with } g.m_0 = m, \tag{6.3.3}$$

where the max is computed component-wise.

Table 6.1: The groups and homogeneous spaces in this work. For each group and each homogeneous space, typical elements are provided, as well as the action of the group element on the space element.

| $G$ \ $\mathcal{M}$ | $\mathbb{R}^3, x$ | $\mathbb{S}^2, \vec{v}$ | $\mathbb{R}^3 \times \mathbb{S}^2, (x, \vec{v})$ |
|---|---|---|---|
| $\mathbb{T}^3, \vec{t}$ | $x + \vec{t}$ | | |
| $SO(3), R$ | | $R\vec{v}$ | |
| $SE(3), (R, \vec{t})$ | $Rx + \vec{t}$ | | $(Rx + \vec{t}, R\vec{v})$ |

**Activation functions and separable kernels.** A point-wise activation function $\alpha$, such as ReLU, is trivially equivariant. On manifolds with a product structure, $\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2$, both for homogeneous spaces and groups, using separable kernels $\kappa = \kappa_{\mathcal{M}_1} \otimes \kappa_{\mathcal{M}_2}$, the layers can be split by sequential application of convolutions on these sub-domains.

**Spaces and groups.** The spaces used in this work are $\mathbb{R}^3$, the sphere $\mathbb{S}^2$ and the product space $\mathbb{R}^3 \times \mathbb{S}^2$. The groups that we consider are: translations of $\mathbb{R}^3$ - $\mathbb{T}^3 \simeq \mathbb{R}^3$, 3D rotations - $SO(3)$, and the special Euclidean group $SE(3) = SO(3) \ltimes \mathbb{T}^3$. Table (6.1) shows the different combinations of spaces and groups. Entries left empty are not used or fail to be homogeneous spaces for standard group actions on them.

### 6.3.2 Discretization of spherical signals

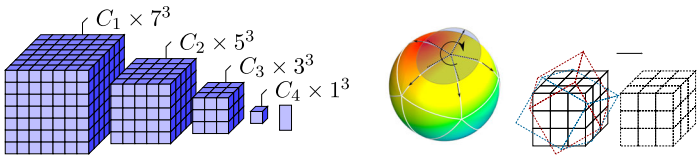
The way spherical signals are numerically handled have major implications for our networks. A DWI signal is treated as a discretization of a signal $f : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. DWIs are acquired, for each voxel, at $N$ fixed directions $p_1, \ldots, p_N$ on $\mathbb{S}^2$ (here $N = 90$) . They can be represented in two different ways.

- Type 1. Ignoring the spherical structure, at each voxel $x$, we get a measurement vector $I(x) = (I(x, p_1) \ldots, I(x, p_N)) \in \mathbb{R}^N$. Thus an image is a mapping $I : \mathbb{R}^3 \to \mathbb{R}^N$.

- Type 2. A signal at voxel $x$ is interpolated as a regular spherical function $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ via $I(x, \vec{v})$ using a Watson kernel [26].

### 6.3.3 Generic Networks used in this work

$\mathbb{T}^3$ The $\mathbb{S}^2$-structure of the signal is ignored, using the Type 1 discretization. The group being $\mathbb{T}^3$, we just obtain a standard CNN, ignoring rotational information. An illustration can be found in Figure 6.1a.




(a) (b) (c)

Figure 6.1: Figure 6.1a is an illustration of the classical CNN. In the illustration, which assembles the true dimensions of the feature maps in the experiment that is presented later, the striding eventually shrinks the grid to a voxel, and features of the voxel are fed into a fully connected layer. Figure 6.1b shows the $SO(3)$ action on $\mathbb{S}^2$. A function (kernel) is moved on $\mathbb{S}^2$ by rotation. Figure 6.1c shows the roto-translational action on $\mathbb{R}^3$, with rotations aligned with those that are part of the $\mathbb{S}^2$ discretization as in Figure 6.1b.

#### 6.3.3.1 $SO(3)$

This time the spatial structure is ignored, thus each voxel provides a spherical data point. Type 2 discretization is used. The GCNN takes as input a spherical function, and will classify it by performing $SO(3)$-lifting, $SO(3)$-convolutions and summarization. The convolved function on $SO(3)$ is then projected back to $\mathbb{S}^2$ by this summarization. It is illustrated in Figure 6.1b.

#### 6.3.3.2 $SE(3)$

Type 2 discretization is used and the network uses the full interplay between spatial roto-translations and corresponding rotations of the spherical signal, and it is separated into a spherical part and a spatial part as explained above. To perform the segmentation task, the projection layer collapses the function on $SE(3)$ back to $\mathbb{R}^3$ by summarizing over $SO(3)$. The spherical part is illustrated in Figure 6.1b, and the spatial part is illustrated in Figure 6.1c. The rotations in Figure 6.1c are aligned with the rotations that moved the spherical kernels in Figure 6.1b.

## 6.4 Experiments and Results

In this section, we first list all the detailed network setups, after which we present the results of the experiments. We evaluate our method on the DWI brain scan dataset from the human connectome project [46]. We classify the human brains into 4 regions - cerebrospinal fluid (CSF), subcortical, white matter, and grey matter. An illustration of the task can be found in Figure 6.2.

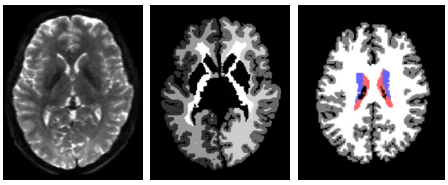
We used the pre-processed DWI data [46] and normalized each DWI scan for the $b$-1000 images with the voxel-wise average of the $b_0$. The labels provided with the T1-image were transformed to the DWI using nearest neighbor inter-

polation (Figure 6.2). Since the 4 brain regions we are classifying have imbalanced numbers of voxels, we use Focal Loss [30] to counter the class imbalance of the dataset. For Focal Loss, all experiments use $\alpha = (0.35, 0.35, 0.15, 0.15)$ for CSF, subcortical, WM, and GM respectively, and $\gamma = 2$. For Watson Kernel, all experiments that used this interpolation (Type 2 discretization) have $\kappa = 10$. Batch size for all experiments is 100.

### 6.4.1 Experiment setup

To reduce the computation burden, as inputting a full DWI volume is intractable, we use spatial windows of $N^3$ voxels, with $N = 1$ for $SO(3)$-action network and $N = 7$ for the rest. In addition, due to the effect of striding in spatial convolution, the $7^3$ grid of voxels shrinks to $1^3$. Therefore, a separable $SE(3)$ convolution layer after this shrinking is equivalent to a single $SO(3)$ convolution layer, since the spatial convolution becomes trivial. $\mathbb{S}^2$ is discretized by a regular icosahedron. $SO(3)$ is discretized as the icosahedral rotation group with 60 elements. Each vertex of the icosahedron is fixed by 5 rotations, isomorphic to the subgroup of $SO(2)$ consisting of rotations of angle $2k\pi/5$, $k = 0 \dots 4$.

#### 6.4.1.1 $\mathbb{T}^3$: Classical CNN



(a) (b) (c)

Figure 6.2: (a)-(c): original diffusion data, the ground-truth segmentation, and the processed ground-truth that we are going to learn from. The label colors for CSF, subcortical, white matter and grey matter are red, blue, white and grey respectively. The figures only illustrate the data, they are not necessarily from the same slice of the same scan.

We use a $\mathbb{R}^3 conv(ReLU) - \mathbb{R}^3 conv(ReLU) - \mathbb{R}^3 conv(ReLU) - FC$ architecture, with a small and a big network setup. We label the small network $(90 - 5 - 5 - 5 - 4)$ Classical$^-$ and the big network $(90 - 120 - 120 - 90 - 4)$ Classical$^+$.

#### 6.4.1.2 $SO(3)$: Baseline

In the experiments, we use the $lift(ReLU) - gconv(ReLU) - project - FC$ architecture as was used in [31], but with true $SO(3)$-convolution. The projection layer takes the maximum of the 5 rotations (fibers) to collapse the function back to the sphere. Two capacities are used - $lift(1) - gconv(5) - FC(4)$ and $lift(10) - gconv(20) - FC(4)$ - resp. named Baseline$^-$ and Baseline$^-$.

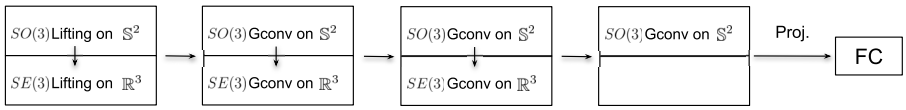



Figure 6.3: Architecture of our $SE(3)$ network. Each block is a convolutional layer split into 2 separable layers. The last block before the FC layer is equivalent to a single $\mathbb{S}^2$-layer as explained in Section 6.4.1.

#### 6.4.1.3 $SE(3)$: Ours

We use the separable setup that was presented above. Thus a layer is again split into 2 layers - an $\mathbb{S}^2$-layer (Figure 6.1b) and an $\mathbb{R}^3$-layer (Figure 6.1c), both for lifting and group convolution. The rotational actions of the kernels can be represented by 60 rotation matrices, and is equivalent to the discretization of the $SO(3)$ rotation group using the icosahedral symmetry group. We use the $lift(ReLU)-gconv(ReLU)-gconv(ReLU)-gconv(ReLU)-project-FC$ architecture. Using the separable convolution explained above, it can be illustrated as in Figure 6.3. We use $5-5-5-5-5-5-5-5-4$ for a small network (Ours$^-$) and $10-20-20-40-40-20-10-4$ for a big network (Ours$^+$).

### 6.4.2 Results

As was done in [31], we trained all networks using **1** scan, validated using **1** scan, and tested using **50** scans.

We evaluate the accuracies and Dice scores of the classification of the 4 regions respectively, and the overall classification accuracy across all test scans. For each class, the accuracy is calculated by $\frac{\#ClassCorrect}{\#ClassSamples}$, and the Dice score by $\frac{2TP}{2TP+FP+FN}$ for the class. The overall accuracy is calculated by $\frac{\#Correct}{\#AllSamples}$.

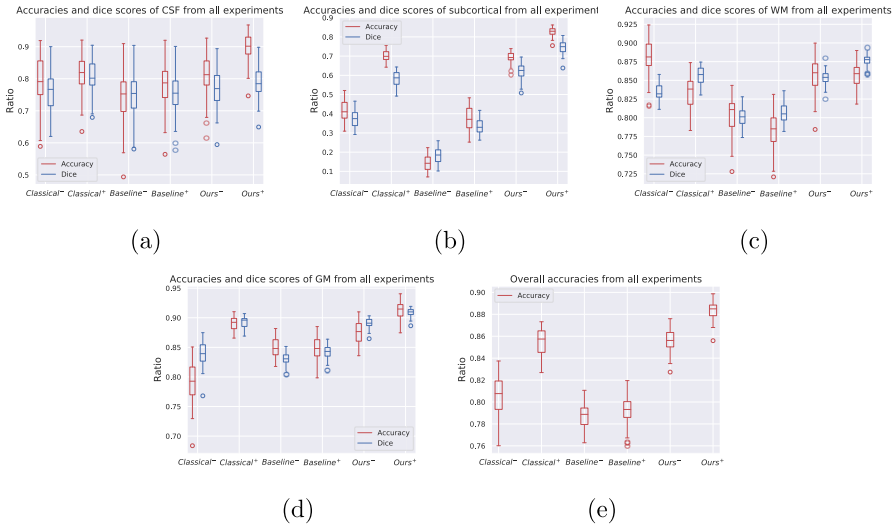
We trained all models until they converge and before overfitting, thus models are stopped at different epochs. Classical$^-$ and Classical$^+$ were trained for 34 and 19 epochs, Baseline$^-$ and Baseline$^+$ were both trained for 31 epochs, and Ours$^-$ and Ours$^+$ were trained for 41 and 15 epochs.

The Dice scores and accuracies of all experiments can be found in Table 6.2. It is easy to see that our method, with the smallest model capacity, has the best performance. The Classical$^+$ setup works well, but it is at the cost of much bigger model capacity. Additionally, Classical$^-$ is not significantly better than Baseline$^+$, even though it has a way larger capacity. Plots of the results can be found in Figure 6.4. Demonstrations of predictions from all models with high capacity can be found in Figure 6.5. Predictions from Ours$^+$ are much less noisy - especially for the subcortical region - than others.

In order to test the model resistance to variations, we rotated the test set

Table 6.2: Statistics of Dice scores and Accuracies.

| G | Model (#Param) | CSF | Subcortical | WM | GM | Overall |
|---|---|---|---|---|---|---|
| **Dice scores** | | | | | | |
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | | |
| $\mathbb{T}^3$ | Classical$^-$(13539) | $0.756 \pm 0.07$ | $0.376 \pm 0.043$ | $0.834 \pm 0.011$ | $0.839 \pm 0.02$ | |
| | Classical$^+$(972694) | $\mathbf{0.804} \pm 0.053$ | $0.583 \pm 0.036$ | $0.856 \pm 0.011$ | $0.893 \pm 0.009$ | |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | | |
| $SO(3)$ | Baseline$^-$(286) | $0.75 \pm 0.073$ | $0.185 \pm 0.04$ | $0.801 \pm 0.012$ | $0.83 \pm 0.011$ | |
| | Baseline$^+$(2104) | $0.754 \pm 0.069$ | $0.334 \pm 0.037$ | $0.805 \pm 0.013$ | $0.841 \pm 0.012$ | |
| $SE(3)$ | Ours$^-$(2514) | $0.769 \pm 0.06$ | $0.621 \pm 0.038$ | $0.854 \pm 0.01$ | $0.891 \pm 0.008$ | |
| | Ours$^+$(59914) | $0.788 \pm 0.05$ | $\mathbf{0.746} \pm 0.034$ | $\mathbf{0.877} \pm 0.008$ | $\mathbf{0.909} \pm 0.006$ | |
| **Accuracies** | | | | | | |
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | | |
| $\mathbb{T}^3$ | Classical$^-$(13539) | $0.792 \pm 0.08$ | $0.415 \pm 0.053$ | $\mathbf{0.879} \pm 0.024$ | $0.789 \pm 0.034$ | $0.806 \pm 0.017$ |
| | Classical$^+$(972694) | $0.815 \pm 0.061$ | $0.702 \pm 0.026$ | $0.834 \pm 0.022$ | $0.89 \pm 0.011$ | $0.854 \pm 0.012$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | | |
| $SO(3)$ | Baseline$^-$(286) | $0.742 \pm 0.082$ | $0.145 \pm 0.04$ | $0.804 \pm 0.024$ | $0.85 \pm 0.016$ | $0.788 \pm 0.011$ |
| | Baseline$^+$(2104) | $0.778 \pm 0.07$ | $0.379 \pm 0.065$ | $0.784 \pm 0.024$ | $0.848 \pm 0.02$ | $0.792 \pm 0.013$ |
| $SE(3)$ | Ours$^-$(2514) | $0.81 \pm 0.065$ | $0.692 \pm 0.029$ | $0.857 \pm 0.022$ | $0.874 \pm 0.019$ | $0.856 \pm 0.01$ |
| | Ours$^+$(59914) | $\mathbf{0.896} \pm 0.042$ | $\mathbf{0.826} \pm 0.023$ | $0.857 \pm 0.017$ | $\mathbf{0.912} \pm 0.014$ | $\mathbf{0.883} \pm 0.008$ |
| **Accuracies from rotated test set** | | | | | | |
| $\mathbb{T}^3$ | Classical$^+$ | $0.611 \pm 0.095$ | $0.494 \pm 0.025$ | $0.506 \pm 0.022$ | $0.551 \pm 0.025$ | $0.53 \pm 0.015$ |
| $SO(3)$ | Baseline$^+$ | $0.769 \pm 0.074$ | $0.307 \pm 0.059$ | $0.782 \pm 0.024$ | $0.846 \pm 0.02$ | $0.786 \pm 0.013$ |
| $SE(3)$ | Ours$^+$ | $\mathbf{0.88} \pm 0.048$ | $\mathbf{0.659} \pm 0.028$ | $\mathbf{0.83} \pm 0.019$ | $\mathbf{0.868} \pm 0.018$ | $\mathbf{0.84} \pm 0.009$ |




(a) (b) (c)

(d) (e)

Figure 6.4: Accuracies and Dice scores of all 4 brain regions individually, and overall accuracy across all experiments.

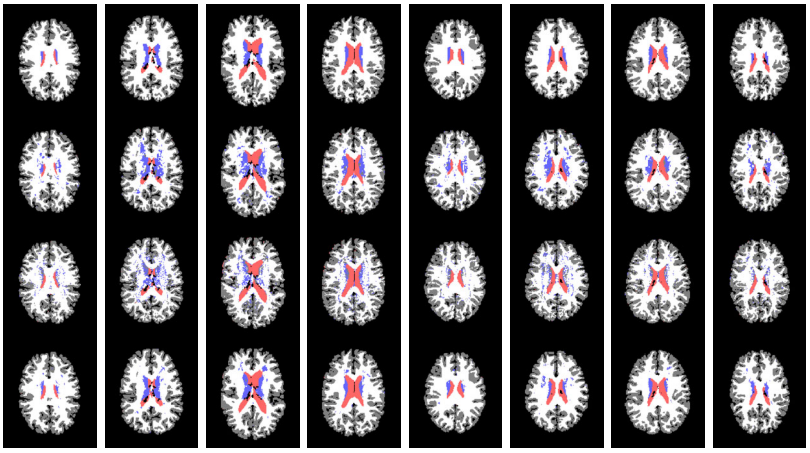


Figure 6.5: From top to bottom are ground-truth, predictions from Classical$^+$, Baseline$^+$, and Ours$^+$. The colors of CSF, subcortical, WM and GM are red, blue, white, and grey respectively.

using rotations randomly sampled from an octahedral rotation group, and the accuracies from the rotated data using high capacity models can be found at the bottom of Table 6.2. Models with rotational group actions (Baseline$^+$ and Ours$^+$) are resistant to data variation, and Ours$^+$ remains the best in performance.

## 6.5 Conclusion

We propose an $SE(3)$ GCNN for DWI scan segmentation by using a natural action of $SE(3)$ on space $\mathbb{R}^3 \times \mathbb{S}^2$, which models the space where DWI data is measured. As it is a homogeneous space for this action, we develop equivariant/invariant GCNNs for functions defined on it. This strategy keeps the required network capacity small, while mitigating the need for data augmentation, which is usually more expensive either in computation or in storage. Experiments show that our method is superior to ones that discard either the spatial symmetries on $\mathbb{R}^3$ or the rotational symmetries on $\mathbb{S}^2$. Additionally, tested with rotated data, models with rotational group actions demonstrate again the impact of equivariance, especially for our $SE(3)$-based model.

# Chapter 7

# A Study on Group Convolutions and Equivariance for DWI Segmentation

## 7.1 abstract

We present a series of Group Convolutional Networks for segmentation of Diffusion Weighted Imaging data. These networks gradually incorporate group actions that are natural for this type of data, in the form of convolutions that provide equivariant transformations of the data. This knowledge provides a potentially important inductive bias and may alleviate the need for data augmentation strategies. We study the effects of these actions on the performances of the networks, by training and validating them using the diffusion data from the Human Connectome project. We show how incorporating more actions generally improves the performances of our segmentation while limiting the number of parameters that must be learned.

## 7.2 Introduction

In this work, we study the influence of group actions on data and how they may impact the architecture and performances of neural networks, especially convolutional neural networks (CNN). CNNs rely on assumed translational symmetries in data and have shown very robust performance in imaging tasks, especially medical imaging ones, and they are highly parameter-efficient thanks to their weight-sharing property. When data offer more structure than simply translation, this can be used to build generalized CNNs. These Group and Geometric CNNs (GCNN) have been studied intensively and applied in

many situations in the few past years ([32, 16, 7, 5, 14] to cite a few). This is especially the case for the task at hand - classification and segmentation of Diffusion Weighted Imaging (DWI) data.

DWI is a non-invasive image modality that provides local information about water diffusion in tissue by means of measuring spins displacement [45]. It provides 3-dimensional diffusion information at each location $x$ that can be encoded as a function $f_x$ on the 2-dimensional sphere $\mathbb{S}^2$. A field of these functions, on a given domain, can be represented as a function $f : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. If a sample is rotated and translated, the acquired signal should reflect, up to the limitations of acquisition protocol, this transformation. The group in question is the group of 3D rigid motions, $SE(3)$, and the space $\mathbb{R}^3 \times \mathbb{S}^2$ is a *homogeneous space* under the action of $SE(3)$: a point in $\mathbb{R}^3 \times \mathbb{S}^2$ can be transformed in any other point by a rigid transformation. This notion of homogeneous space is at the heart of the extension of CNNS to GCNNs [14, 6].

Our task at hand is the classification/segmentation of diffusion data. The inductive bias provided by the knowledge of these transformations may prove important for our task, especially when the amount of annotated data is limited. The problem boils down to how to incorporate this knowledge. The most classical approach is to use data augmentation, reflecting the expected symmetries in the data, in the hope that the network will be able to learn it during the training phase, learning symmetry-aware kernels.

Incorporating, on the other hand, some information about the symmetries of the data in the model has been shown to boost the performances of these networks [5]. But how much of this information is needed for a given task? To provide an answer, for the DWI segmentation task, we propose several networks, which gradually incorporate these symmetries in their architecture and study their performances. They, in some sense, perform a *group action ablation study*. We start with a "naive" CNN, then incorporate spherical symmetries, resulting in a $SO(3)$-GCNN, discarding the spatial aspect of the data. The spatial aspect is then added in the form of a standard CNN coupled with spherical symmetries and then a network where roto-translational transformations are used in almost all steps. This work demonstrates empirically the improvement in performances. The results are, however, not always clear-cut. The GCNN built from 3D-translations on one hand and rotations on the other hand seems to perform better than a $SE(3)$-GCNN. However, the $SE(3)$-network generalizes better to unseen rotated data than the previous one. The reason may lie in the particular type of data used - our DWI scans come from the Human Connectome Project (HCP) [46] are highly preprocessed, including a form of alignment, and this may impact the results.

In the rest of this paper, we review related work, both around CNN and DWI classification problem. Then we introduce the theoretical setup of GCNN. We build several networks. We then study and discuss their performances.

## 7.3 Related Work

Deep Learning (DL) for non-flat data, or using more complex group actions than just translations, is currently getting more attention from the research field. When it comes to non-flat data, such as the point-wise spherical signals in DWI, particularly relevant related works are the following. Masci *et al.* [32] proposed a NN on surfaces that extracts local rotationally invariant features. A non-rotationally invariant modification was proposed by Boscaini *et al.* [7]. The above provide methods for DL-based processing of data on arbitrary manifolds. When the manifold, however, is a homogeneous space, i.e., there is a group action by which any two points on the manifolds can be reached, theory simplifies via a natural generalization of classical convolutions in group convolution neural networks (GCNNs) [15, 5, 28]. GCNNs guarantee global equivariance. However, global equivariance can be complicated and elusive when the underlying geometry is non-trivial [44]. An elementary construction on a general manifold is proposed by Schonsheck *et al.* [37] via a fixed choice of geodesic paths used to transport filters between points on the manifold, ignoring the effects of path dependency, i.e. holonomy when paths are geodesics. The removal of this path dependency can be obtained by summarizing local responses over local orientations, which is what was done by Masci *et al.* [32]. To explicitly deal with holonomy, Sommer *et al.* [42] proposed a theoretical breakthrough using convolution construction on manifolds based on stochastic processes via the frame bundle.

On the other hand, Cohen *et al.* [15] lifted spherical functions to the 3D-rotation group $SO(3)$ and used a generalization of Fourier transform on it to perform convolution. With the generalization of convolution to more complex group actions than translation, several authors [22, 16, 47, 50, 28, 5, 1, 9, 10, 11] explored the group convolution path for Lie groups and the homogeneous spaces of these groups. The relation between group actions, principal bundles and related vector bundles, and convolutional architectures is currently explored [44, 14, 2]. The latter elicidates important relations between differential geometry of bundles and Reproducible Kernel Hilbert Spaces. Links between partial differential equations, symmetries and GCNN is studied in [41]. A unifying framework for equivariant DL on manifolds, connecting both the bundle and homogeneous space viewpoint, is given in [49] through a notion of coordinate indepencent convolutions.

Most CNNs approach for the processing of DWI signals discard its specific structure. For instance, Golkov *et al.* [24] built multi-layer perceptrons in $q$-space for kurtosis and NODDI mappings. However, the importance of spherical equivariant or invariant structure has been acknowledged for some years now. The importance of the extraction of rotationally invariant features beyond Fractional Anisotropy [4] has been recognized in series of DWI works. For instance, Caruyer *et al.* [8] developed invariant polynomials of spherical harmonic (SH) expansion coefficients, and discussed their application in

population studies. Schwab *et al.*[38] proposed a related construction using eigenvalue decomposition of SH operators. Novikov et al. [35] and Zucchelli *et al.* [52] argued their usefulness for understanding microstructures in relation to DWI.

Chakraborty *et al.* [9] proposed a rotation equivariant construction inspired by Cohen *et al.* [15] for disease classification. The same authors [3] used a $\mathbb{S}^2 \times \mathbb{R}^+$ CNN using SHORE function representation for classification in Parkinson Disease. Sedlar *et al.* [40] used a spherical U-Net for f-ODF estimation The same authors [39] used a spherical CNN for microstructure parameter estimation, using spherical harmonics representations. Müller *et al.* [34] propose a sixth-D, 3D space and $q$-space NNs with roto-translation / rotation equivalence properties.

## 7.4 Method

The networks we present will be built from the principle of expanding CNNs to groups $G$ and homogeneous spaces $\mathcal{M}$, on which they act by extending convolution operations to functions on groups and their homogeneous spaces. However, we do not follow the common path of irreducible representations for implementing convolutions/correlations over $\mathbb{S}^2$ or $SO(3)$.

An action of a Lie group on a space $\mathcal{M}$ is, for our purpose a smooth mapping $G \times \mathcal{M} \to \mathcal{M}$, $(g, m) \to g.m$ such that for each $g$, $m \to g.m$ is a diffeomorphism of $\mathcal{M}$ and such that $g.(g'.m) = (gg').m$. The neutral element of $G$ acts as the identity. The orbit of $m \in \mathcal{M}$ is the set $G.m = \{g.m, g \in G\}$. The stabilizer $G_m$ of an element $m$ is the set of transformations that lets $m$ fixed, $G_m = \{g \in G, g.m = m\}$. It is a subgroup of $G$. $\mathcal{M}$ is a *$G$-homogeneous space* if it contains only one orbit, i.e, if for any $m, m' \in \mathcal{M}$, there exists $g \in G$, with $g.m = m'$. Given a $m_0$ in the homogeneous space $\mathcal{M}$, there is an isomorphism $G/G_{m_0} \simeq \mathcal{M}$, called the orbit map. $G/G_{m_0}$ is the quotient space of $G$ by $G_{m_0}$ and consists of the *left cosets* $gG_{m_0}$ of $G_{m_0}$. The inverse of the point $m$ by the orbit map is a coset $gG_{m_0}$, with $g.m_0 = m$, called the *fiber* above $m$.

### 7.4.1 Standard convolution operations

A group $G$ acting on a space $\mathcal{M}$ via $(g, m) \mapsto g.m$ also acts on functions on $\mathcal{M}$ by the *left translation*

$$(L_g f)(m) = f(g^{-1}m). \tag{7.4.1}$$

We assume that each homogeneous space is endowed with a $G$-invariant measure that allows integration, and that each $G$ is endowed with a left-invariant Haar measure.

#### 7.4.1.1 Lifting layer

A function $f : \mathcal{M} \to \mathbb{R}^N$ can be *lifted* to the group $G$ via a kernel $\kappa : \mathcal{M} \to \mathbb{R}^K$ by

$$\kappa * f(g) = \sum_{i=1}^{K} \int_{\mathcal{M}} f(m)\kappa_i(g^{-1}m)\, dm \tag{7.4.2}$$

This operation is *equivariant*: $\kappa * L_g f = L_g(\kappa * f)$.

#### 7.4.1.2 Group convolution layer

A feature function $F : G \to \mathbb{R}^N$ can be transformed by a convolution kernel $K : G \to K$ by

$$K * F(g) = \sum_{i=1}^{N} \int_{G} F(h)K_i(h^{-1}g)dh. \tag{7.4.3}$$

This operation is equivariant: $K * (L_g F) = L_g(K * F)$.

#### 7.4.1.3 Projection Layer

If needed, feature map $F : G \to \mathbb{R}^n$ can be projected to a function $f : \mathcal{M} \to \mathbb{R}^n$ by summarizing on the fibers

$$\overline{F}(m) = \max_{h \in G_{m_0}} F(gh), \quad \text{for any } g \text{ with } g.m_0 = m, \tag{7.4.4}$$

where the max is computed component-wise. This operation is equivariant: $\overline{L_k F} = L_k \overline{F}$.

#### 7.4.1.4 Activation Functions and Separable Kernels

A point-wise activation function $\alpha$, such as ReLU, is trivially equivariant $L_g(\alpha f) = \alpha L_g f$. On manifolds with an underlying product structure, $\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2$ - this includes homogeneous spaces and groups - one can choose separable kernels $\kappa = \kappa_{\mathcal{M}_1} \otimes \kappa_{\mathcal{M}_2}$, and activation functions can be introduced in (7.4.2) and (7.4.3). For instance, lifting (7.4.2) can be replaced by

$$\kappa *^{\alpha} f(g) = \sum_{i=1}^{K} \int_{\mathcal{M}_1} \alpha\left(\int_{\mathcal{M}_2} f(m_1, m_2)\kappa_2(g^{-1}m_2)\, dm_2\right) \kappa_1(g^{-1}m_1)\, dm_1, \tag{7.4.5}$$

which preserves equivariance. This is used in this work.

The spaces used in this work are $\mathbb{R}^3$, the sphere $\mathbb{S}^2$ and the product space $\mathbb{R}^3 \times \mathbb{S}^2$. The groups that we consider are the group of translations of $\mathbb{R}^3$, $\mathbb{T}^3 \simeq \mathbb{R}^3$, the group $SO(3)$ or 3D rotations, the direct product $\mathcal{G} = \mathbb{T}^3 \times SO(3)$

| $G$ \ $\mathcal{M}$ | $\mathbb{R}^3, x$ | $\mathbb{S}^2, \vec{v}$ | $\mathbb{R}^3 \times \mathbb{S}^2, (x, \vec{v})$ |
|---|---|---|---|
| $\mathbb{T}^3, \vec{t}$ | $x + \vec{t}$ | | |
| $SO(3), R$ | | $R\vec{v}$ | |
| $\mathbb{T}^3 \times SO(3), (\vec{t}, R)$ | | | $(x + \vec{t}, R\vec{v})$ |
| $SE(3), (R, \vec{t})$ | $Rx + \vec{t}$ | | $(Rx + \vec{t}, R\vec{v})$ |

Table 7.1: The groups and homogeneous spaces in this work. For each group and each homogeneous space, typical elements are provided, as well as the action of the group element on the space element.

and the special Euclidean group $SE(3) = SO(3) \ltimes \mathbb{T}^3$. Note that though $\mathcal{G}$ and $SE(3)$ are isomorphic as manifolds, they are not as groups: in $\mathcal{G}$, $(\vec{t}, R).(\vec{s}, S) = (\vec{t} + \vec{s}, RS)$ while in $SE(3)$, $(R, \vec{t}).(S, \vec{s}) = (RS, \vec{t} + R\vec{s})$. This is also reflected in their respective actions in Table (7.1), which shows the different combinations of spaces and groups. Entries left empty are not used or fail to be homogeneous spaces for standard group actions on them.

### 7.4.2 Pseudo-convolutions on $\mathbb{S}^2$

In our previous work [31], we proposed another way of filtering signals on $\mathbb{S}^2$. Instead of lifting a signal $f : \mathbb{S}^2 \to \mathbb{R}^n$ to $SO(3)$, we lift it, above each point $q$ of $\mathbb{S}^2$ to the rotations that let $q$ fixed, via a set of localized spherical kernels transported along predetermined paths from a given base-point to each point where we analyze our signal. This, in fact, means that the lifted space is isomorphic to $\mathbb{S}^2 \times SO(2)$. Then convolutions are performed on each fibre, independently of each other and a local pooling is performed to get the information back on $\mathbb{S}^2$, before being fed to a fully convolutional network for classification. A predetermined transport is performed by moving the kernel from a base point $p_0$ to any point $q$, via a transformation $\sigma_q : \sigma_q \cdot p_0 = q$. For equivariance to hold, either the kernel is rotationally invariant, which is very restrictive, or one should have $\sigma_{Rq} = R\sigma_q$ for any $R \in SO(3)$, which cannot hold. Thus equivariance does not hold in general. Details are found in [31]. This is different from our baseline network, which imposes $SO(3)$-equivariance.

### 7.4.3 Discretization of spherical signals

The way spherical signals are numerically handled have major implications for our networks. A DWI signal is treated as a discretization of a signal $f : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. DWIs are acquired, for each voxel, at $N$ fixed directions $p_1, \ldots, p_N$ on $\mathbb{S}^2$ (here $N = 90$) . These are represented in two different ways.

- Type 1. Ignoring the spherical structure, at each voxel $x$, we get a measurement vector $f(x) = (f(x, p_1) \ldots, f(x, p_N)) \in \mathbb{R}^N$. Thus an image is a mapping $I : \mathbb{R}^3 \to \mathbb{R}^N$.

- Type 2. A signal at voxel $x$ is interpolated as a proper spherical function $f(x, \vec{v}) = W(v; v_1, \ldots, v_N)$ where $W$ is a Watson kernel [26]. An image from this type is a mapping $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$.

### 7.4.4 Generic Networks used in this work

We present 4 constructions in which gradual levels of complexity in group actions are introduced. This can be seen as a group-action ablation study. The precise description of each network will be provided in Section 7.5.

#### 7.4.4.1 $\mathbb{T}^3$

The $\mathbb{S}^2$-structure of the signal is ignored, using the Type 1 discretization. The group being $\mathbb{T}^3$, we just obtain a standard CNN, ignoring rotational information. An illustration can be found in Figure 7.1.

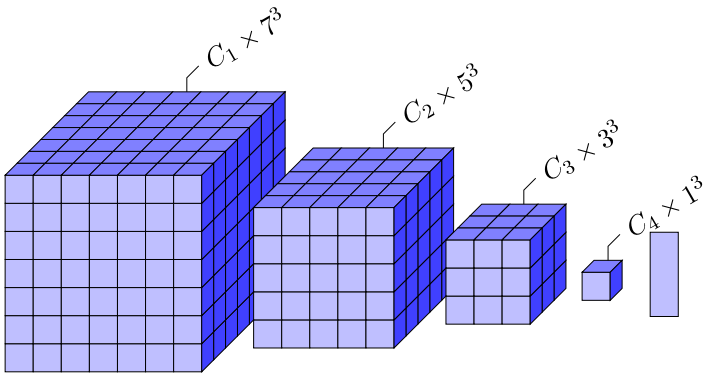
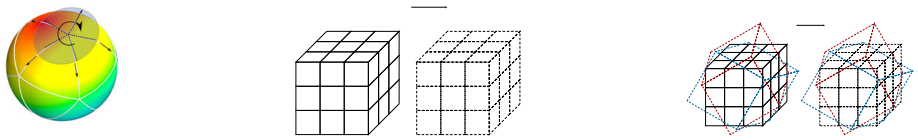



Figure 7.1: Illustration of the classical CNN. In the grids shown above, which assembles the dimensions of feature maps in the later experiments. Each voxel in the *ith* layer contains $C_i$ values, indicating the numbers of channels. $C_1$ here is the number of signal values each voxel from the original scan, thus 90. Due to striding, the grid shrinks to 1 voxel after 3 convolutional layers, and then is fed into a fully connected layer for classification.

#### 7.4.4.2 $SO(3)$

This time the spatial structure is ignored, and each voxel provides a spherical data point. Type 2 discretization is used. The GCNN takes as input a spherical function, and will classify it by performing $SO(3)$-lifting, $SO(3)$-convolutions and summarization. The convolved function on $SO(3)$ is then projected back to $\mathbb{S}^2$ by this summarization. It is illustrated in Figure 7.2a.



(a) (b) (c)

Figure 7.2: Figure 7.2a illustrates the GCNN kernel of $SO(3)$ action. A function (kernel) is moved on $\mathbb{S}^2$ by rotation. Figure 7.2b illustrates the translational action $\mathbb{T}^3$ on $\mathbb{R}^3$. Figure 7.2c shows the roto-translational action on $\mathbb{R}^3$, with rotations aligned with those that are part of the $\mathbb{S}^2$ discretization as in Figure 7.2a. Figure 7.2a and Figure 7.2b together are the 2 decoupled layers of a $\mathbb{T}^3 \times SO(3)$-action convolutional layer, while Figure 7.2a and Figure 7.2c together are the 2 separable layers of an $SE(3)$-action layer.

#### 7.4.4.3 $\mathbb{T}^3 \times SO(3)$

Spatial and spherical structures are decoupled. This implies a standard spatial CNN dealing with only voxel translations, and a $SO(3)$-GCNN part for the directional signal. Type 2 discretization is used for spherical signals. The decoupled $\mathbb{R}^3$-layer and $\mathbb{S}^2$-layer with group actions $\mathbb{T}^3$ and $SO(3)$ respectively can be found in Figure 7.2b and Figure 7.2a. Note that since the spatial convolution does not incorporate rotation equivariance, it does not reflect equivariance of the DWI measurements. I.e., one can expect that when the brain rotates, the spatial patterns rotate as well as their spherical diffusion signals. This model takes rotation into account in the spherical part of the signal, but not the spatial part. The projection at the end collapses the function in the group back to $\mathbb{R}^3$ by summarizing - in this case, maximizing - over $SO(3)$, and the resulting feature map is fed into a fully connected layer to perform the classification task.

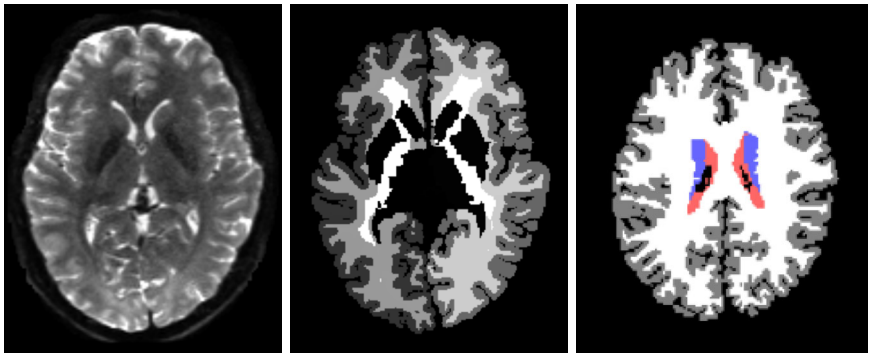
#### 7.4.4.4 $SE(3)$

Type 2 discretization is used and the network uses the full interplay between spatial roto-translations and corresponding rotations of the spherical signal and is thus fully equivariant to $SE(3)$ transformations on the DWI data. Figure 7.2a shows for kernel for the $\mathbb{S}^2$-layer. When the kernel moves from one vertex to another, it follows a specific rotation that maps the one-ring neighborhood of the source vertex to the one-ring neighborhood of the target vertex. At each vertex, the kernel has an $SO(2)$ symmetry group structure discretized by 5 rotations. Figure 7.2c shows the kernel for the $\mathbb{R}^3$-layer. It is rotated with the same rotation matrices that moved the $\mathbb{S}^2$-kernel as in Figure 7.2a. Again, to perform the segmentation task, the projection layer collapses the function on $SE(3)$ back to $\mathbb{R}^3$ by summarizing - again, maximizing - over $SO(3)$.

## 7.5 Experiments and Results

In this section, we first list all the detailed network setups, after which we present the results of the experiments. We evaluate our method on the DWI brain dataset from the human connectome project (HCP) [46]. We classify the human brains into 4 regions - cerebrospinal fluid (CSF), subcortical, white matter (WM), and grey matter (GM). An illustration of the task can be found in Figure 7.3.

We use the pre-processed DWI data [46] and normalize each DWI scan for the $b$-1000 images with the voxel-wise average of the $b_0$. The labels provided with the T1-image are transformed to the DWI using nearest neighbor interpolation (Figure 7.3b). Focal Loss [30] is used to counter the class imbalance of the 4 brain regions. For Focal Loss, all experiments use $\gamma = 2$ and use $\alpha = (0.35, 0.35, 0.15, 0.15)$ for CSF, subcortical, WM, and GM respectively. For the Watson Kernel, all experiments that used this interpolation (Type 2 discretization) have $\kappa = 10$. Batch size for all experiments is 100.



(a) (b) (c)

Figure 7.3: (a)-(c): original diffusion data, the ground-truth segmentation, and the processed ground-truth that we are going to learn from. The label colors for CSF, subcortical, white matter and grey matter are red, blue, white and grey respectively. The figures are only for illustrations of the data, they are not necessarily from the same slice of the same scan.

### 7.5.1 Experiment setup

To reduce the computational burden, as inputting a full DWI volume is intractable, we use spatial windows of $N^3$ voxels, with $N = 1$ for the $SO(3)$-action network and $N = 7$ for the rest. In addition, due to the effect of striding in spatial convolution, the $7^3$ grid of voxels shrinks to $1^3$ after 3 spatial convolutions. Therefore, a separable convolution layer (for both $\mathbb{T}^3 \times SO(3)$ and $SE(3)$ actions) is equivalent to a single $SO(3)$ convolution layer when the grid

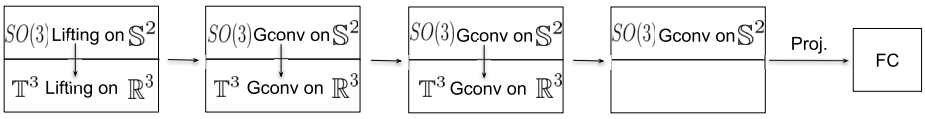



Figure 7.4: Architecture of the network with group action $\mathbb{T}^3 \times SO(3)$. Each block is a convolutional layer split into 2 separable layers. The last block before the FC layer is equivalent to a single $\mathbb{S}^2$-layer as explained in Section 7.5.1. Illustrations of ReLU actions are omitted for visualization simplicity.

shrinks to $1^3$, since the spatial convolution becomes trivial. $\mathbb{S}^2$ is discretized by a regular icosahedron. $SO(3)$ is discretized as the icosahedral rotation group with 60 elements. Each vertex of the icosahedron is fixed by 5 rotations, isomorphic to the subgroup of $SO(2)$ consisting of rotations of angle $2k\pi/5$, $k = 0 \dots 4$. This is, of course, the discretization used for $SO(2)$.

#### 7.5.1.1 $\mathbb{T}^3$: Classical CNN

We use a $\mathbb{R}^3 conv(ReLU) - \mathbb{R}^3 conv(ReLU) - \mathbb{R}^3 conv(ReLU) - FC$ architecture with network setups of a low capacity and a high capacity. We label the small network $(90-5-5-5-4)$ Classical$^-$ and the big network $(90-120-120-90-4)$ Classical$^+$.

#### 7.5.1.2 $SO(3)$-Baseline

In the experiments, we use the $lift(ReLU) - gconv(ReLU) - project - FC$ architecture as was used in [31], but with true $SO(3)$-convolution. The projection layer takes the maximum of the 5 rotations to collapse the function back to the sphere. We experimented various sizes of the network $(10-20-proj.-4$ and $20-40-proj.-4)$, in addition to the setup used in [31] $(1-5-proj.-4)$. The network that has the biggest size did not seem to improve the second biggest one thus we omit it in this paper. Based on the size of the experiments, we call the small network Baseline$^-$ and the big network Baseline$^+$.

#### 7.5.1.3 $\mathbb{T}^3 \times SO(3)$-OursDecoupled

We use the architecture $lift(ReLU) - gconv(ReLU) - gconv(ReLU) - gconv(ReLU) - project - FC$. Using separability discussed in Section 7.4.1.4, a convolution layer (including lifting) is split into 2, and ReLU activation is added between separable layers as well. An illustration of the architecture can be found in Figure 7.4.

We again experiment with 2 sizes of the network - a small one and a big one. The small network has $5-5-5-5-5-5-5-proj.-4$ kernels for each layer, while the big network has $10-20-20-40-40-20-10-proj.-4$. We label them OursDecoupled$^-$ and OursDecoupled$^+$.

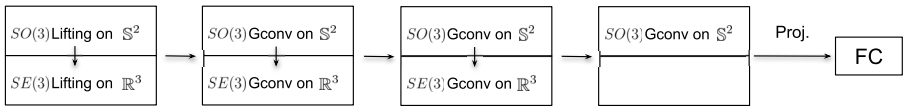



Figure 7.5: Architecture of the network with group action $SE(3)$.

#### 7.5.1.4 $SE(3)$-Ours

Here too we use the separable setup described in Section 7.4.1.4. Thus a layer is again split into 2 layers - an $\mathbb{S}^2$-layer and an $\mathbb{R}^3$-layer, both for lifting and group convolution. The $\mathbb{S}^2$-layer is defined as shown in Figure 7.2a. We rotate the $\mathbb{R}^3$ kernels and the $\mathbb{S}^2$ kernels using the same actions. The rotational actions of the kernels can be represented by 60 rotation matrices, and is equivalent to the discretization of the $SO(3)$ rotation group using the icosahedral symmetry group, as shown in Figure 7.2c. As in Section 7.5.1.3, we use the $lift(ReLU) - gconv(ReLU) - gconv(ReLU) - gconv(ReLU) - project - FC$ architecture. After the separation of the layers, the illustration is showcased in Figure 7.5. As in Section 7.5.1.3, ReLU activations are added between separable layers as well.

In addition, we intend to explore the impact of the equivariance we imposed in $\mathbb{R}^3$ in this section. As was explained above, we align the rotations of the $\mathbb{R}^3$ kernel with the ways the $\mathbb{S}^2$ kernel moved on the sphere, which is discretized by the 60 rotation symmetries of an icosahedron. At a vertex $x_i, i \in 1, ..., 12$ of an icosahedron, there exists a stabilizer $SO(2)_{x_i}$ discretized by 5 equally divided rotations that keep $x_i$ unchanged. Therefore, we also experiment a partial equivariance in the $\mathbb{R}^3$ roto-translational convolution. This means at each vertex $x_i$ of the icosahedron, we only take 1 out of the 5 rotations that discretized $SO_{x_i}(2)$ instead of using all of them to rotate the spatial kernel. Note that the Part models are only fully $SE(3)$ equivariant when the kernels have a sub-group $SO(2)$ symmetry in them [6, Thm 1], which we do not impose and thus equivariance is not guaranteed.

Again, we experiment with 2 sizes of the network with $5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - proj. - 4$ and $10 - 20 - 20 - 40 - 40 - 20 - 10 - proj. - 4$ kernels respectively. Therefore, we generate 4 experiments for this section: OursFull$^-$, OursPart$^-$, OursFull$^+$, and OursPart$^+$.

A summary of the experiments can be found in Table 7.2.

### 7.5.2 Results

As was done in [31], we trained all networks using **1** scan, validated using **1** scan, and tested using **50** scans. We evaluate the accuracies and Dice scores of the classification of the 4 regions respectively, and the overall classification accuracy across all test scans. For each class, the accuracy is calculated by

| Experiment | $G$ | #Params | #Epochs |
|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | |
| Classical$^-$ | $\mathbb{T}^3$ | 13539 | 34 |
| Classical$^+$ | | 972694 | 19 |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | |
| Baseline$^-$ | $SO(3)$ | 286 | 31 |
| Baseline$^+$ | | 2104 | 31 |
| OursDecoupled$^-$ | $\mathbb{T}^3 \times SO(3)$ | 2514 | 41 |
| OursDecoupled$^+$ | | 59914 | 15 |
| OursPart$^-$ | $SE(3)*$ | 2514 | 41 |
| OursPart$^+$ | | 59914 | 15 |
| OursFull$^-$ | $SE(3)$ | 2514 | 41 |
| OursFull$^+$ | | 59914 | 15 |

Table 7.2: Criteria and properties of experiments. $SE(3)*$ indicates the rotations in the spatial part are only a part of the rotations used in the spherical part.

| Experiment / Class | CSF | Subcortical | WM | GM |
|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | |
| Classical$^-$ | $0.756 \pm 0.07$ | $0.376 \pm 0.043$ | $0.834 \pm 0.011$ | $0.839 \pm 0.02$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | |
| Baseline$^-$ | $0.75 \pm 0.073$ | $0.185 \pm 0.04$ | $0.801 \pm 0.012$ | $0.83 \pm 0.011$ |
| OursDecoupled$^-$ | $\mathbf{0.817} \pm 0.051$ | $\mathbf{0.705} \pm 0.033$ | $\mathbf{0.867} \pm 0.009$ | $\mathbf{0.909} \pm 0.007$ |
| OursPart$^-$ | $0.807 \pm 0.048$ | $0.658 \pm 0.037$ | $0.865 \pm 0.009$ | $0.899 \pm 0.008$ |
| OursFull$^-$ | $0.769 \pm 0.06$ | $0.621 \pm 0.038$ | $0.854 \pm 0.01$ | $0.891 \pm 0.008$ |

Table 7.3: Statistics of Dice scores from experiments using models of low capacity.

$\frac{\#CorrectPredictions}{\#ClassSamples}$, and the Dice score is calculated by $\frac{2TP}{2TP+FP+FN}$ for the class. The overall accuracy is calculated by $\frac{\#CorrectPredictions}{\#AllSamples}$.

We trained all models until they converge and before overfitting, thus models of different capacities and different setups are stopped at different epochs. Details can be found in Table 7.2.

The Dice scores and accuracies of models of low capacity can be found in Table 7.3 and Table 7.4, while the Dice scores and accuracies of models of high capacity can be found in Table 7.5 and Table 7.6. Examples of predictions compared with the ground-truth can be found in Figure 7.9a.

#### 7.5.2.1 The impact of the $\mathbb{R}^3$ spatial component

It is easy to observe that the the Baseline experiments perform the worst among all. This is an anticipated outcome since it is usually the case that

| Experiment / Class | CSF | Subcortical | WM | GM | Overall |
|---|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | |
| Classical$^-$ | $0.792 \pm 0.08$ | $0.415 \pm 0.053$ | $\mathbf{0.879} \pm 0.024$ | $0.789 \pm 0.034$ | $0.806 \pm 0.017$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | |
| Baseline$^-$ | $0.742 \pm 0.082$ | $0.145 \pm 0.04$ | $0.804 \pm 0.024$ | $0.85 \pm 0.016$ | $0.788 \pm 0.011$ |
| OursDecoupled$^-$ | $\mathbf{0.844} \pm 0.061$ | $\mathbf{0.741} \pm 0.033$ | $0.833 \pm 0.02$ | $\mathbf{0.934} \pm 0.013$ | $\mathbf{0.878} \pm 0.009$ |
| OursPart$^-$ | $0.787 \pm 0.068$ | $0.717 \pm 0.032$ | $0.848 \pm 0.019$ | $0.906 \pm 0.016$ | $0.868 \pm 0.009$ |
| OursFull$^-$ | $0.81 \pm 0.065$ | $0.692 \pm 0.029$ | $0.857 \pm 0.022$ | $0.874 \pm 0.019$ | $0.856 \pm 0.01$ |

Table 7.4: Statistics of classification accuracy from all experiments using models of low capacity.

neighboring information is an essential type of local features.

#### 7.5.2.2 Type 1 discretization vs Type 2 discretization

The classical CNNs use Type 1 discretization while Type 2 discretization is used for the rest of the models. The classical CNNs do not perform as well as models that take into account the spherical geometry with spatial information, but performs better than Baseline. However, Classical$^-$ is not much better than Baseline$^+$ while having far more parameters to train, and Classical$^+$ performs even worse than OursDecoupled$^-$, OursPart$^-$, or OursFull$^-$, which have much less training parameters.

The results of the two extreme cases - Baseline that only takes into account but ignore any spatial information and Classical that only looks into the spatial part and discards spherical geometry - show that the voxel geometry and neighboring voxel correlation can both capture some decent amount of information to deal with the segmentation task, but they both have something that the other one cannot grasp, and combining the spherical geometry and the spatial correlation can boost the performance to a promising extent.

#### 7.5.2.3 The impact of adding an $\mathbb{R}^3$ part to Baseline

On top of the Baseline, the easiest way to add spatial information to the purely voxel-based framework is what was done in OursDecoupled Section 7.5.1.3 - a GCNN on $\mathbb{S}^2$ to learn the geometric signals in individual signals and a regular classical CNN to take into account the local spatial information. We can see from the results that this setup immediately boosted the performance compared to the Baseline. We can also see that OursDecoupled$^+$ performs better than OursDecoupled$^-$, for the sake of model capacity.

#### 7.5.2.4 The argument for OursFull not performing the best

For models of low capacity, however, we can observe from Table 7.3 and Table 7.4 that our proposed method performs worse than OursDecoupled$^-$. Ad-

| Experiment / Class | CSF | Subcortical | WM | GM |
|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | |
| Classical$^+$ | $0.804 \pm 0.053$ | $0.583 \pm 0.036$ | $0.856 \pm 0.011$ | $0.893 \pm 0.009$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | |
| Baseline$^+$ | $0.754 \pm 0.069$ | $0.334 \pm 0.037$ | $0.805 \pm 0.013$ | $0.841 \pm 0.012$ |
| OursDecoupled$^+$ | $0.827 \pm 0.047$ | $0.716 \pm 0.044$ | $0.878 \pm 0.009$ | $0.903 \pm 0.01$ |
| OursPart$^+$ | $\mathbf{0.834} \pm 0.045$ | $\mathbf{0.752} \pm 0.034$ | $\mathbf{0.878} \pm 0.009$ | $\mathbf{0.914} \pm 0.007$ |
| OursFull$^+$ | $0.788 \pm 0.05$ | $0.746 \pm 0.034$ | $0.877 \pm 0.008$ | $0.909 \pm 0.006$ |

Table 7.5: Statistics of Dice scores from experiments using models of high capacity.

| Experiment / Class | CSF | Subcortical | WM | GM | Overall |
|---|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | |
| Classical$^+$ | $0.815 \pm 0.061$ | $0.702 \pm 0.026$ | $0.834 \pm 0.022$ | $0.89 \pm 0.011$ | $0.854 \pm 0.012$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | |
| Baseline$^+$ | $0.778 \pm 0.07$ | $0.379 \pm 0.065$ | $0.784 \pm 0.024$ | $0.848 \pm 0.02$ | $0.792 \pm 0.013$ |
| OursDecoupled$^+$ | $0.865 \pm 0.061$ | $0.783 \pm 0.035$ | $\mathbf{0.867} \pm 0.017$ | $0.902 \pm 0.019$ | $0.879 \pm 0.011$ |
| OursPart$^+$ | $0.819 \pm 0.065$ | $0.816 \pm 0.031$ | $0.845 \pm 0.019$ | $\mathbf{0.936} \pm 0.011$ | $\mathbf{0.888} \pm 0.009$ |
| OursFull$^+$ | $\mathbf{0.896} \pm 0.042$ | $\mathbf{0.826} \pm 0.023$ | $0.857 \pm 0.017$ | $0.912 \pm 0.014$ | $0.883 \pm 0.008$ |

Table 7.6: Statistics of classification accuracy from experiments using models of high capacity.

ditionally, for models of high capacity, even though we can see that OursFull$^+$ and OursPart$^+$ improve from their low capacity counterparts more than OursDecoupled$^+$, OursFull$^+$ does not perform as well as OursPart$^+$ as shown in Table 7.5 and Table 7.6. This differs from our expectation since models with full roto-translational equivariance should be more capable of handling variances in data, thus should have better performance. Recall that the HCP dataset[46] contains scans that are preprocessed and aligned with axes, thus there is little variance in rotation. In this case, enforcing $SE(3)$ equivariance in the model can be futile and be even confusing for the model.

To verify this theory, we evaluated all models on augmented data. Taking the $N^3$ ($N = 1$ for Baseline models and $N = 7$ for the rest) grids of voxels we extracted from the test scans, we randomly rotate each grid using rotations sampled from the octahedral symmetry group to create a new augmented test set. In this way, we do not need to interpolate while rotating, and the rotations are not aligned with the ones we used in our models to rotate the kernels while still resemble a discretization of the $SO(3)$ group.

#### 7.5.2.5 Experiments with synthetically rotated test set

For models with both low and high capacity, OursFull models have the best performance among other models. OursFull$^-$ remains 0.823 accuracy, decreased from 0.856 while OursFull$^+$ decreased from 0.883 to 0.84. This is

| Experiment / Class | CSF | Subcortical | WM | GM |
|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | |
| Classical$^-$ | $0.603 \pm 0.103$ | $0.127 \pm 0.015$ | $0.706 \pm 0.015$ | $0.57 \pm 0.039$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | |
| Baseline$^-$ | $0.735 \pm 0.076$ | $0.158 \pm 0.037$ | $0.799 \pm 0.013$ | $0.829 \pm 0.011$ |
| OursDecoupled$^-$ | $0.708 \pm 0.073$ | $0.531 \pm 0.033$ | $0.801 \pm 0.012$ | $0.851 \pm 0.006$ |
| OursPart$^-$ | $0.714 \pm 0.069$ | $\mathbf{0.536} \pm 0.035$ | $0.804 \pm 0.011$ | $0.851 \pm 0.008$ |
| OursFull$^-$ | $\mathbf{0.737} \pm 0.065$ | $0.517 \pm 0.033$ | $\mathbf{0.823} \pm 0.01$ | $\mathbf{0.867} \pm 0.009$ |

Table 7.7: Statistics of dice scores from experiments using rotated data and models of small capacity.

| Experiment / Class | CSF | Subcortical | WM | GM | Overall |
|---|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | |
| Classical$^-$ | $0.645 \pm 0.109$ | $0.282 \pm 0.045$ | $0.781 \pm 0.04$ | $0.436 \pm 0.044$ | $0.579 \pm 0.02$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | |
| Baseline$^-$ | $0.733 \pm 0.085$ | $0.12 \pm 0.035$ | $0.802 \pm 0.024$ | $0.852 \pm 0.016$ | $0.786 \pm 0.011$ |
| OursDecoupled$^-$ | $0.755 \pm 0.076$ | $0.528 \pm 0.037$ | $0.779 \pm 0.02$ | $\mathbf{0.871} \pm 0.013$ | $0.81 \pm 0.008$ |
| OursPart$^-$ | $0.69 \pm 0.084$ | $\mathbf{0.599} \pm 0.033$ | $0.791 \pm 0.02$ | $0.852 \pm 0.018$ | $0.809 \pm 0.009$ |
| OursFull$^-$ | $\mathbf{0.79} \pm 0.067$ | $0.591 \pm 0.026$ | $\mathbf{0.835} \pm 0.023$ | $0.84 \pm 0.022$ | $\mathbf{0.823} \pm 0.01$ |

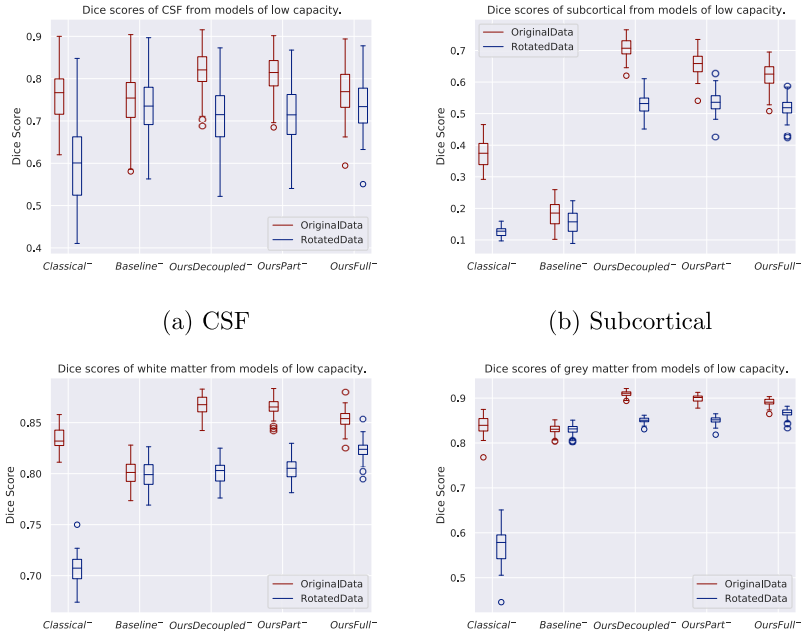
Table 7.8: Statistics of classification accuracy from experiments using rotated data and models of low capacity.

| Experiment / Class | CSF | Subcortical | WM | GM |
|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | |
| Classical$^+$ | $0.54 \pm 0.105$ | $0.169 \pm 0.01$ | $0.59 \pm 0.015$ | $0.617 \pm 0.018$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | |
| Baseline$^+$ | $0.733 \pm 0.076$ | $0.282 \pm 0.036$ | $0.799 \pm 0.013$ | $0.839 \pm 0.012$ |
| OursDecoupled$^+$ | $0.702 \pm 0.075$ | $0.497 \pm 0.037$ | $0.8 \pm 0.011$ | $0.829 \pm 0.009$ |
| OursPart$^+$ | $0.734 \pm 0.063$ | $0.58 \pm 0.033$ | $0.806 \pm 0.011$ | $0.862 \pm 0.006$ |
| OursFull$^+$ | $\mathbf{0.74} \pm 0.06$ | $\mathbf{0.604} \pm 0.034$ | $\mathbf{0.835} \pm 0.01$ | $\mathbf{0.877} \pm 0.008$ |

Table 7.9: Statistics of dice scores from experiments using rotated data and models of high capacity.

| Experiment / Class | CSF | Subcortical | WM | GM | Overall |
|---|---|---|---|---|---|
| $I : \mathbb{R}^3 \to \mathbb{R}^N$ | | | | | |
| Classical$^+$ | $0.611 \pm 0.095$ | $0.494 \pm 0.025$ | $0.506 \pm 0.022$ | $0.551 \pm 0.025$ | $0.53 \pm 0.015$ |
| $I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ | | | | | |
| Baseline$^+$ | $0.769 \pm 0.074$ | $0.307 \pm 0.059$ | $0.782 \pm 0.024$ | $0.846 \pm 0.02$ | $0.786 \pm 0.013$ |
| OursDecoupled$^+$ | $0.756 \pm 0.082$ | $0.597 \pm 0.034$ | $0.797 \pm 0.019$ | $0.81 \pm 0.019$ | $0.791 \pm 0.01$ |
| OursPart$^+$ | $0.716 \pm 0.078$ | $0.635 \pm 0.033$ | $0.78 \pm 0.021$ | $\mathbf{0.876} \pm 0.012$ | $0.819 \pm 0.008$ |
| OursFull$^+$ | $\mathbf{0.88} \pm 0.048$ | $\mathbf{0.659} \pm 0.028$ | $\mathbf{0.83} \pm 0.019$ | $0.868 \pm 0.018$ | $\mathbf{0.84} \pm 0.009$ |

Table 7.10: Statistics of classification accuracy from experiments using rotated data and models of high capacity




(a) CSF

(b) Subcortical

(c) White matter

(d) Grey matter

Figure 7.6: Comparison of Dice scores of the 4 classes from the original and rotated data using models of low capacity.

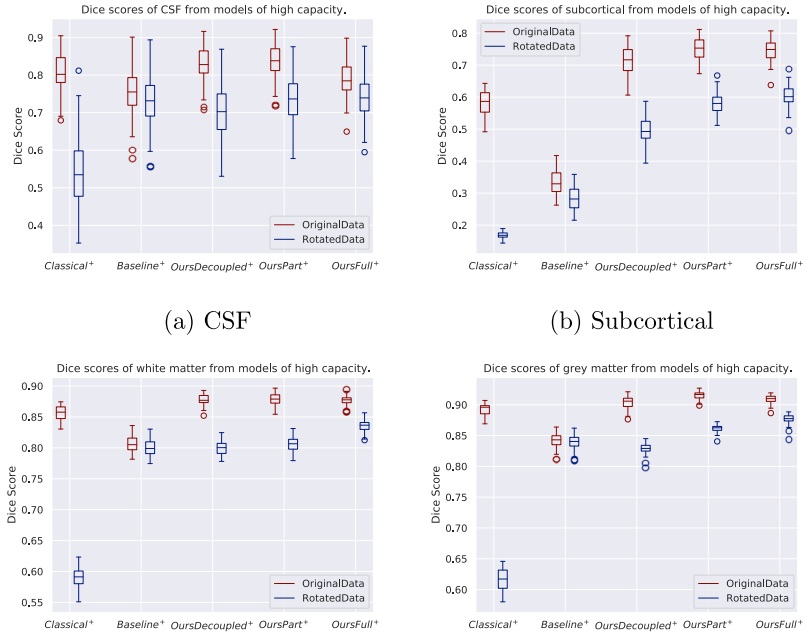
illustrated in Table 7.8 and Table 7.10. In terms of Dice scores, OursFull$^-$ performs the best for all classes but the subcortical class, and OursFull$^+$ has the best results for **all** classes, as shown in Table 7.7 and Table 7.9.

Comparison figures of the 4 classes for models with both low and high capacity can be found in Figure 7.6 and Figure 7.7, while comparisons of overall accuracies can be found in Figure 7.8. We can see again from the model with no spatial equivariance (OursDecoupled), the model with partial spatial equivariance (OursPart), and the model with full spatial equivariance (OursFull) that the gap between the performances on original data and rotated data shrink.

It is worth noticing that Baseline models almost do not suffer from performance drop while applied with rotated data. It is an $SO(3)$ network that preserves rotational equivariance on $\mathbb{S}^2$. For a single-voxel input, the network is very resistant to variations, but the performance of this model is limited due to the lack of spatial interaction and thus in general worse than models with spatial interplay.

Examples of predictions using the rotated test set can be found in Figure 7.9b. It is easily observed that the classical CNN does not generalize well

to the data variation, while models with rotational symmetry (either $SO(3)$, $\mathbb{T}^3 \times SO(3)$, or $SE(3)$) generate better results. However, it is also noticeable that for a challenging minority class, subcortical region, OursFull$^+$ performs better than the others while other models with some rotational equivariance do not predict a concentrated subcortical region. Zoom-in examples can be found in Figure 7.11. Predictions from Baseline are omitted from Figure 7.11 since it does not have the same level of performance.




(a) CSF

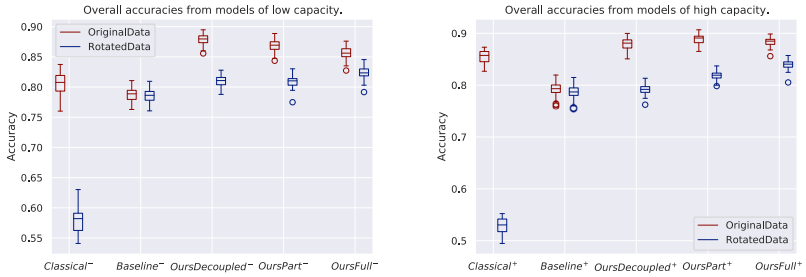
(b) Subcortical

(c) White matter

(d) Grey matter

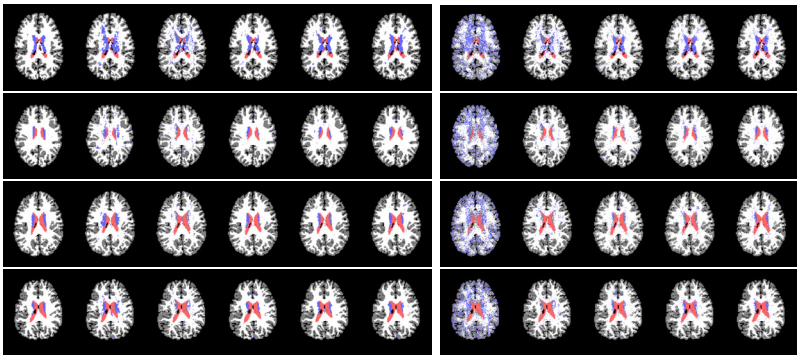
Figure 7.7: Comparison of dice scores of the 4 classes from the original and rotated data using models of high capacity.

We can see clearly from Figure 7.10 as well that the performance of classical CNN decreases the most using rotated data, and the decrease of performance goes down when we enforce more spatial equivariance in the model. Baseline models decrease the least, but again, the performance is limited due to the lack of information in $\mathbb{R}^3$. Furthermore, the $SE(3)$ equivariance is implemented separately for the spatial and spherical parts, and is with interpolation in the spatial part, thus there are some errors introduced to it. Therefore, OursFull models always perform the best when there is variation in the data.




(a) Low capacity models (b) High capacity models

Figure 7.8: Comparison of overall accuracies from the original and rotated data.

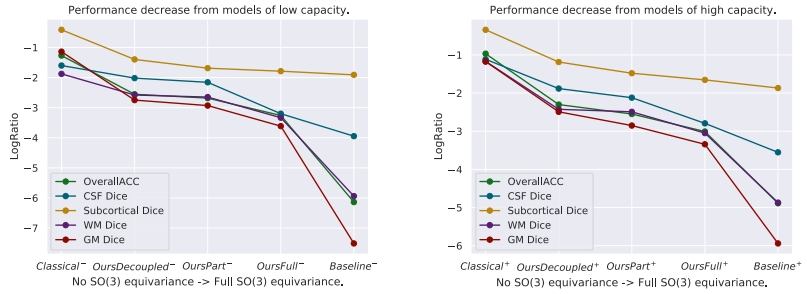


(a) Predictions from test set using original data. (b) Predictions from test set using rotated data.

Figure 7.9: Examples of predictions. Figure 7.9a shows the predictions from the original test set, and Figure 7.9b shows the predictions from the augmented (rotated) test set. In Figure 7.9a, from left to right are ground-truth, Classical$^+$, Baseline$^+$, OursDecoupled$^+$, OursPart$^+$, and OursFull$^+$. In Figure 7.9b, from left to right are Classical$^+$, Baseline$^+$, OursDecoupled$^+$, OursPart$^+$, and OursFull$^+$. The colors of CSF, subcortical, WM and GM are red, blue, white, and grey respectively.

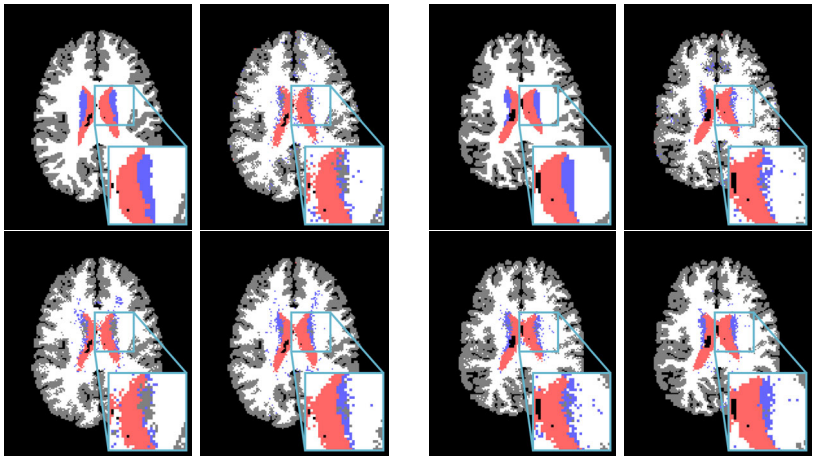
#### 7.5.2.6 Rotational invariance for Type 1 discretization

Further more, we have also experimented with networks that have some rotational invariance but in the classical CNN setup - viewing the DWI images as $I : \mathbb{R}^3 \to \mathbb{R}^N$. Taking the classical CNN setup we have in Section 7.5.1.1, we rotate the CNN kernels in each layer using the same rotations as in Section 7.5.1.4 to discretize $SO(3)$. As was done above, we use the 60 rotations from the icosahedral symmetry group as well as only 12 of them (1 at each rotation axis) to act on the CNN kernels. In each layer, one rotation of the




(a) Low capacity models (b) High capacity models

Figure 7.10: Logarithm of the ratio of decrease of performances using rotated data compared to using original data.



(a) A test scan slice. (b) Another test scan slice.

Figure 7.11: Showcases of zoom-in regions from predictions of the rotated test set. For both scan slices presented, from left to right, top to bottom, are the ground-truth, prediction from OursDecoupled$^+$, OursPart$^+$, and OursFull$^+$.

kernel is only convolved with the response of the corresponding rotation from the last layer, thus this network is in fact 60 (or 12) independent networks, in which they share the same weights of different rotations. At the end, we take the average of the 60 (or 12) responses from all the rotations. With a small trial, we discovered that, as expected, even though this type of network does not perform as well as our spatial-directional GCNN as a whole, the performance decreases little in the full icosahedral group case with 60 rotations when tested with augmented data, and decreases more when only a subset (12) of the group is used to rotate the kernels. See Table 7.11.

This further explains that having rotational equivariance in the model

| Rotations | Data Type | CSF Dice | Subcortical Dice | WM Dice | GM Dice | Overall ACC |
|---|---|---|---|---|---|---|
| $90-5-5-5-FC$, #Param 13539 | | | | | | |
| Part(12) | Original | $0.798 \pm 0.058$ | $0.425 \pm 0.052$ | $0.843 \pm 0.01$ | $0.875 \pm 0.01$ | $0.838 \pm 0.011$ |
| | Rotated | $0.71 \pm 0.074$ | $0.306 \pm 0.042$ | $0.755 \pm 0.014$ | $0.796 \pm 0.014$ | $0.75 \pm 0.013$ |
| Full(60) | Original | $0.754 \pm 0.065$ | $0.485 \pm 0.059$ | $0.823 \pm 0.014$ | $0.848 \pm 0.02$ | $0.818 \pm 0.016$ |
| | Rotated | $0.75 \pm 0.063$ | $0.479 \pm 0.059$ | $0.813 \pm 0.013$ | $0.838 \pm 0.02$ | $0.809 \pm 0.016$ |

Table 7.11: Augmented CNN tested with original and rotated data.

makes it much more robust to variance in the data - which, with no need of explanation, is inevitable when dealing with real-world raw data. Averaging rotational copies of a classical CNN achieves the goal of dealing with variance in data, but for data formats like DWI of which signals in voxels have some geometric structure, our full $SE(3)$ GCNN provides the best solution.

## 7.6 Discussion

The resistance to data variation that has been shown by our fully equivariant network was demonstrated on synthetically augmented data - with 90-degree rotations. Even though this synthetic augmentation did not cost any loss of signals or any interpolation-caused inaccuracy, it is desirable to verify the robustness of more complex group actions in CNNs using data with real-world variations (e.g. subjects scanned in different positions). Acquiring this type of data is another challenge.

## 7.7 Conclusion

We presented a systematic study of GCNNs of various group actions with the application to DWI segmentation. We interpreted images of DWI scans ($I : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$) as functions in the homogeneous spaces of groups with different complexities of symmetries and provided a detailed analysis of how different levels of complexities of these symmetries impact the performance of the network. From the experiments, we conclude that 1) exploiting the spatial-directional interactions in the data is crucial for efficient learning of the features; 2) incorporating complex group actions of 3D rigid motions - SE(3) - might not be essential for highly aligned and preprocessed data like the human connectome project (HCP) [46], but it shows significantly higher resistance to variations in data. For real-world raw data in which positions of subjects are not perfectly aligned as in [46], our proposal shows great potential.

# Chapter 8

# Discussion and Future Work

We proposed a series of works generalizing CNNs to more complex group actions (and their symmetries) than translation, mainly in the application of DWI segmentation. We first proposed a framework that treats individual voxels as functions on a general manifold, lifting functions locally to a 2D rotation group. The pseudo-convolutions are then performed in these local groups, after which we summarize the rotations to obtain local rotational invariance. This framework breaks equivariance, yet the elimination of path dependency of parallel transport provides us with robust features that allow us to showcase performances at the same level as the state-of-the-art. We then presented models in which true group convolutions are performed. The data were measured on the homogeneous spaces $(\mathbb{R}^3, \mathbb{S}^2, \mathbb{R}^3 \times \mathbb{S}^2)$ of the groups that they were lifted to $(\mathbb{T}^3, SO(3), \mathbb{T}^3 \times SO(3), SE(3))$. We experimented with cases where 1) the spherical part of the data is discarded; 2) the spatial part of the data is discarded; 3) both spherical and spatial aspects are taken into account in the modeling. From the comparison of the three cases, we have observed the uniquely important role each part plays in performing the task, and the combination of both spherical and spatial aspects gives us the best performance. Additionally, for case 3), we experimented with different types of actions in the spatial part in which we gradually incorporated more rotational equivariance in the model: a) translation action; b) roto-translation in which the rotations are partly aligned with the spherical rotation actions; c) roto-translation that has fully aligned rotation actions with the spherical part. From the experiments, we observe that case c) does not seem to bring better performance than a) and b). One possible cause for this result is that the type of data we used is highly processed and is with a form of alignment, such that the enforced rotational equivariance in the spatial aspect might actually have little effect. Therefore, we synthetically rotated the test set and tested all the models on it. The results showed that with more equivariance imposed in the model, the more resistant the model is to this data variation.

Even though the synthetically rotated data are generated using random

samples from an octahedral rotation group that does not introduce interpolation-induced artifacts, it is desirable in the future, though, to verify the resistance of our proposed methods on real-world raw data with no manual alignment. Acquiring this type of data, of course, becomes another challenge.

Moreover, having established generalized CNNs with natural group actions according to the data type, we can apply them to other data modalities, e.g. geo-spatial data, satellite data, etc, in that they have data structures that cannot be fully modeled simply by translation action and that the analyses of these data have potential in aiding a wide range of tasks that were mostly done manually. It is, in like manner, desirable to perform different tasks on diffusion data other than segmentation, e.g. tracking fibers in a brain.

# Bibliography

[1] Andrearczyk, V., Fageot, J., Depeursinge, A.: Local Rotation Invariance in 3D CNNs. Medical Image Analysis **65** (2020)

[2] Aronsson, J.: Homogeneous Vector Bundles and $\mathcal{G}$-Equivariant Convolutional Neural Networks (2021)

[3] Banerjee, M., Chakraborty, R., Archer, D., Vaillancourt, D., Vemuri, B.C.: DMR-CNN: A CNN Tailored for DMR Scans with Applications to PD Calssifiaction. In: Proceedings of International Symposium on Biomedical Imaging (2019)

[4] Basser, P., Mattiello, J., LeBihan, D.: MR Diffusion Tensor Spectroscopy and Imaging. Biophys. J. **66**(1), 259–267 (1994)

[5] Bekkers, E., Veta, M.L.M., Eppenhof, K., Pluim, J., Duits, R.: Roto-Translation Covariant Convolutional Networks for Medical Image Analysis. In: Proc. MICCAI 2018. pp. 440–448 (2018)

[6] Bekkers, E.J.: B-spline cnns on lie groups. In: International Conference on Learning Representations (2019)

[7] Boscaini, D., Masci, J., Rodolà, E., Bronstein, M.: Learning Shape Correspondence With Anisotropic Convolutional Neural Networks. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 29 (2016)

[8] Caruyer, E., Verma, R.: On Facilitating the Use of HARDI in Population Studies by Creating Rotation-Invariant Markers. Medical Image Analysis **20**(1), 87–96 (2015)

[9] Chakraborty, R., Banerjee, M., Vemuri, B.: A CNN for Homogeneous Riemannian Manifolds with Application to NeuroImaging (2018)

[10] Chakraborty, R., Banerjee, M., Vemuri, B.C.: H-cnns: Convolutional neural networks for riemannian homogeneous spaces. arXiv preprint arXiv:1805.05487 **1** (2018)

[11] Chakraborty, R., Bouza, J., Manton, J., Vemuri, B.C.: Manifoldnet: A deep neural network for manifold-valued data with applications. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)

[12] Cheng, H., Newman, S., M. Afzali, S.S.F., Garyfallidis, E.: Segmentation of the Brain Using Direction-Averaged Signal of DWI Images. Magnetic Resonance Imaging **69**, 1–7 (2020)

[13] Cohen, T.S., Welling, M.: Steerable CNNs. arXiv e-prints (dec 2016)

[14] Cohen, T., Geiger, M., Weiler, M.: A general theory of equivariant cnns on homogeneous spaces (2020)

[15] Cohen, T., Geiger, M., Köhler, J., Welling, M.: Spherical CNNs. In: International Conference on Learning Representations (2018)

[16] Cohen, T., Welling, M.: Group equivariant convolutional neural networks. In: Int. Conf. Machine Learning. pp. 2990–2999 (2016)

[17] de Figueiredo, E.H., Borgonovi, A.F., Doring, T.M.: Basic concepts of mr imaging, diffusion mr imaging, and diffusion tensor imaging. Magnetic Resonance Imaging Clinics of North America **19**(1), 1–22 (2011). https://doi.org/https://doi.org/10.1016/j.mric.2010.10.005, `https://www.sciencedirect.com/science/article/pii/S1064968910000747`, diffusion Imaging: From Head to Toe

[18] Diestel, J., Spalsbury, A.: The joys of haar measure (2014)

[19] Do Carmo, M.P.: Riemannian Geometry. Birkhauser (1992)

[20] Driscoll, J., Healy, D.: Computing Fourier Transforms and Convolutions on the 2-sphere. Adv. Appl. Math **15**(2) (1994)

[21] Gallier, J., Quaintance, J.: Differential Geometry and Lie Groups, A Scond Course. No. 13 in Geometry and Computing

[22] Gens, R., Domingos, P.: Deep Symmetry networks. In: NIPS. pp. 2537––2545 (2014)

[23] Gerken, J.E., Aronsson, J., Carlsson, O., Linander, H., Ohlsson, F., Petersson, C., Persson, D.: Geometric deep learning and equivariant neural networks (2021)

[24] Golkov, V., Dosovitskit, A., Sperl, J.I., Menzel, M.I., Czisch, M., Särmann, P., Brox, T., Cremers, D.: $q$-Space Deep Learning: Twelve-Fold Shorter and Model-Free Diffusion MRI Scans. IEEE Trans. Med. Im. **35**(5), 1344–1351 (2016)

[25] Henmar, S., Simonsen, E.B., Berg, R.W.: What are the gray and white matter volumes of the human spinal cord? Journal of Neurophysiology **124**(6), 1792–1797 (2020). https://doi.org/10.1152/jn.00413.2020, pMID: 33085549

[26] Jupp, P.E., Mardia, K.V.: A Unified View of the Theory of Directional Statistics, 1975-1988. International Statistical Review / Revue Internationale de Statistique **57**(3), 261–294 (1989)

[27] Klein, A., Andersson, J., Ardekani, B.A., Ashburner, J., Avants, B., Chiang, M.C., Christensen, G.E., Collins, D.L., Gee, J., Hellier, P., et al.: Evaluation of 14 nonlinear deformation algorithms applied to human brain mri registration. Neuroimage **46**(3), 786–802 (2009)

[28] Kondor, R., Trivedi, S.: On the Generalization of Equivariance and Convolution in Neural Networks to the Action of Compact Groups. In: Proc. ICML. pp. 2747–2755 (2018)

[29] Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet Classification with Deep Convolutional Neural Networks **60**(6), 84–90

[30] Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal Loss for Dense Object Detection (2018)

[31] Liu, R., Lauze, F., Erleben, K., Darkner, S.: Bundle geodesic convolutional neural network for dwi segmentation from single scan learning. In: Cetin-Karayumak, S., Christiaens, D., Figini, M., Guevara, P., Gyori, N., Nath, V., Pieciak, T. (eds.) Computational Diffusion MRI. pp. 121–132. Springer International Publishing, Cham (2021)

[32] Masci, J., Boscaini, D., Bronstein, M., Vandergheynst, P.: Geodesic Convolutional Neural Networks on Riemannian Manifolds. In: Proceeding of 3dRR (2015)

[33] Merriam-Webster: Diffusion coefficient, `https://www.merriam-webster.com/dictionary/diffusion%20coefficient`

[34] Müller, P., Golkov, V., Tomassini, V., Cremers, D.: Rotation-Equivariant Deep Learning for Diffusion MRI (2021)

[35] Novikov, D., Veraart, J., Jelescu, I., Fieremans, E.: Rotationally-Invariant Mapping of Scalar and Orientational Metrics of Neuronal Microstructure with Diffusion MRI. NeuroImage **174**, 518–538 (2018)

[36] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge (2015)

[37] Schonsheck, S.C., Dong, B., Lai, R.: Parallel Transport Convolution: A New Tool for Convolutional Neural Networks on Manifolds (2018)

[38] Schwab, E., Cetingül, H.E., Asfari, B., Vidal, E.: Rotational Invariant Features for HARDI. In: Proc. IPMI (2013)

[39] Sedlar, S., Alimi, A., Papadopoulo, T., Deriche, R., Deslauriers-Gauthier, S.: A Spherical Convolutional Neural Network for White Matter Structure Imaging via dMRI. In: de Bruijne, M., Cattin, P.C., Cotin, S., Padoy, N., Speidel, S., Zheng, Y., Essert, C. (eds.) Medical Image Computing and Computer Assisted Intervention – MICCAI 2021. pp. 529–539. Springe (2021)

[40] Sedlar, S., Papadopoulo, T., Deriche, R., Deslauriers-Gauthier, S.: Diffusion MRI fiber orientation distribution function estimation using voxel-wise spherical U-net. In: International MICCAI Workshop 2020 - Computational Diffusion MRI. Lima, Peru (Oct 2020), `https://hal.archives-ouvertes.fr/hal-02946371`

[41] Smets, B.M.N., Portegies, J., Bekkers, E.J., Duits, R.: PDE-based Group Equivariant Convolutional Neural Networks (2021)

[42] Sommer, S., Bronstein, A.: Horizontal Flows and Manifold Stochastics in Geometric Deep Learning. IEEE Trans. PAMI (2020)

[43] Sotiropoulos, S., Moeller, S., Jbabdi, S., Xu, J., Andersson, J.L., Auerbach, E.J., Yacoub, E., Feinberg, D., Setsompop, K., Wald, L., Behrens, T.E.J., Ugurbil, K., Lenglet, C.: Effects of Image Reconstruction on Fiber Orientation Mapping From Multichannel Diffusion MRI: Reducing the Noise Floor Using SENSE. Magnetic Resonance in Medicine **70**(6), 1682 – 1689 (2013)

[44] T.S. Cohen and M. Weiler and B. Kicanaoglu and M. Welling: Gauge equivariant convolutional networks and the icosahedral cnn. In: Proc. ICML. pp. 1321–1330 (2019)

[45] Tuchs, D.S.: Q-Ball Imaging. Magnetic Resonance in Medicine **52**, 1358–1372 (2004)

[46] Van Essen, D.C., Smith, S.M., Barch, D.M., Behrens, T.E., Yacoub, E., Ugurbil, K.: The WU-Minn Human Connectome Project: An Overview. NeuroImage **80**, 62 – 79 (2013)

[47] Weiler, M., Geiger, M., Welling, M., Boomsma, W., Cohen, T.: 3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data. In: Proc. NIPS (2018)

[48] Weiler, M., Hamprecht, F., Storath, M.: Learning Steerable Filters for Rotation Equivariant Cnns. In: Proc. CVPR. pp. 849–858 (2018)

[49] Weiler, M., Forré, P., Verlinde, E., Welling, M.: Convolutional networks–isometry and gauge equivariant convolutions on riemannian manifolds. arXiv preprint arXiv:2106.06020 (2021)

[50] Worrall, D., Garbin, S., Turmukhambetov, D., Brostow, G.: Harmonic Networks: Deep Translation and Rotation Equivariance (2017)

[51] Yap, P.T., Zhang, Y., Shen, D.: Brain Tissue Segmentation Based on Diffusion MRI Using $\ell 0$ Sparse-Group Representation Classification. In: Proc. MICCAI - III. pp. 132–139 (2015)

[52] Zucchelli, M., Deslauriers-Gauthier, S., Deriche, R.: A Computational Framework for Generating Rotation Invariant Features and its Application in Diffusion MRI. Medical Image Analysis **60** (2020)