A Higher Order Manifold-valued Convolutional Neural Network with Applications to Diffusion MRI Processing

Jose J. Bouza[1⋆], Chun-Hao Yang[1⋆[0000−0002−2522−5957]], David Vaillancourt[2], and Baba C. Vemuri[1⋆⋆]

[1] University of Florida CISE, Gainesville, FL 32611, US
[2] University of Florida Applied Physiology and Kinesiology, Gainesville, Fl 32611, US

Abstract. In this paper, we present a novel generalization of the Volterra Series, which can be viewed as a higher-order convolution, to manifold-valued functions. A special case of the manifold-valued Volterra Series (MVVS) gives us a natural extension of the ordinary convolution to manifold-valued functions that we call, the manifold-valued convolution (MVC). We prove that these generalizations preserve the equivariance properties of the Euclidean Volterra Series and the traditional convolution operator. We present novel deep network architectures using the MVVS and the MVC operations which are then validated via two experiments. These include, (i) movement disorder classification from diffusion magnetic resonance images (dMRI), and (ii) fiber orientation distribution function (fODF) reconstruction from compressed sensed dMRIs. In both the experiments, MVVS and MVC networks outperform the state-of-the-art.

Keywords: Riemannian manifolds · Volterra series · convolutional neural network · diffusion MRI · fODF reconstruction · Geometric deep learning

1 Introduction

Theory In the recent past, there has been a surge in medical imaging and computer vision research to develop deep neural networks(DNNs) that can cope with manifold-valued data e.g., the manifold of $(n \times n)$ symmetric positive-definite (SPD) matrices, P_n, the special orthogonal group, $\mathsf{SO}(n)$, the Grassmann manifold, $\mathsf{Gr}(p, n)$, and the n-sphere, S^n. At the outset, it will be useful to categorize two types of problems concerning data in non-Euclidean spaces. These are: (i) data that are samples of functions defined on smooth manifolds, i.e. $f : M \to R$ and (ii) data that are samples of manifold-valued functions whose domain is Euclidean i.e., $f : Z^d \to M$ where M is a Riemannian manifold and Z^d is a Euclidean sample lattice. *In this paper we address the problem of developing DNNs for the data type defined in (ii).*

For methods suited to data in category (i) described above, we refer the reader to a recent survey [5]. In the context of data described in (ii) above, authors in [15] presented a DNN that consists of layers which explicitly utilize the structure of SPD matrices. In [16] authors presented a DNN for classification of hand-crafted features residing in

⋆ These authors contributed equally to the work presented here.

⋆⋆ Email: vemuri@ufl.edu. This research was in part funded by the NSF grant IIS-1724174 to BCV; This paper will appear in the Proc. of IPMI 2021.

a Grassmann manifold. However, the above architectures do not attempt to develop a counterpart of the classical convolutional layer in the traditional convolutional neural network (CNN) which is viewed as one of the key components to the success of CNNs. Besides convolutional layers, batch normalization is also a useful trick used in CNNs to smooth the loss surface, and authors in [6] recently proposed such a technique for data in the manifold of SPD matrices. In this paper, we focus our attention to data represented on a grid where each grid point is associated with a value in a manifold, M, with known geometry, i.e., $f : Z^d \to M$. The lack of a consistent framework for designing DNN architectures for data residing in a general Riemannian manifold was partly due to the fact that unlike for functions defined on manifolds, there was no natural analog of the convolution operation for manifold-valued functions until recently. In [24] authors defined the weighted Fréchet mean (wFM) [23] as an analog to the classical (Euclidean space) convolution operation for manifold-valued data and recently, the use of wFM operation to build a CNN for manifold-valued data was pioneered by authors of [7, 8]. Note that although their definition of wFM as a "plug-in" operation for convolutions is valid for any Riemannian manifold, the convexity constraints in the definition used for wFM restricts the range of values of the wFM leading to model capacity limitations of their network.

In this paper, we propose the idea that for *complete Riemannian manifolds*, it is possible to map the manifold-valued data points within a convolution window defined over the manifold-valued image to the tangent space anchored at the FM of these points using the Riemannian **Log** map. Then, perform the linear combination operation in the tangent space (which is isomorphic to the Euclidean space) and map it back to the manifold using the Riemannian **Exp** map. We provide the details of this operation called the manifold-valued convolution (MVC) in the next section. To increase the expressiveness and hence the capacity of the network, we introduce the novel concept of higher order manifold-valued convolutions via Volterra series representation [26]. The traditional convolution is indeed the first order term of the Volterra Series, which will be briefly reviewed in Section 2. In [18], authors empirically showed that replacing a convolution filter with a higher order Volterra series filter increased model accuracy. The Volterra Series was also used recently to design DNNs for data in category (i) [3]. In this paper, we generalize the Volterra Series for real-valued functions to manifold-valued functions and call it the manifold-valued Volterra Series (MVVS). We show that the MVVS (MVC) is equivariant to translation in the domain which allows for weight sharing. The MVVS (MVC) can be used as an alternative to the wFM-based convolutions presented in [7, 8] and we call the network based on MVVS (MVC) the MVVS (MVC)-Net. In addition to the translation equivariance, the MVC is also equivariant to the isometry group actions admitted by the manifold. This latter equivariance however does not hold for the MVVS. Hence, by considering only the first-order term of the MVVS, we lose some expressiveness, but we gain the isometry equivariance and computational efficiency. Note that the MVC and the wFM-based convolution are different by construction and a key difference is that for wFM the associated weights need to be positive while such restriction is not required by the MVC. In practice, this restriction limits the output of a wFM layer to the convex cone of the input data points and hence greatly reduces the capacity of the network.

Applications To demonstrate the performance of the MVVS (MVC)-Net, we test the proposed network on classification and reconstruction problems encountered in diffusion magnetic resonance image (dMRI) processing. In the context of classification, we apply MVVS and MVC networks to classify dMRI brain scans of patients with movement disorders from controls. In the context of reconstruction, we will reconstruct the fiber orientation distribution function (fODF) field [29] from highly undersampled dMRI data. There is a vast body of literature on fODF reconstruction from dMRI data and we refer the reader to a recent comprehensive survey [11] and references therein. Here, we limit ourselves to the review of DNNs for fODF reconstruction from compressed sensed dMRI data. Recently, authors in [27] proposed a novel deep spherical U-Net for the fODF reconstruction but did not enforce non-negativity constraint on the reconstructed fODFs. They represent the fODF in terms of the spherical harmonics (SH) and the reconstruction thus involves estimating the SH coefficients. In [20,21] 3D-CNN networks were explored for fODF reconstruction, but these networks do not guarantee the non-negativity of the reconstructed fODFs. We choose to use the square-root parametrization of the fODF which maps fODFs to a hypersphere. *Since all operations in our network are intrinsic, the output is automatically a valid (non-negative) fODF*. In fODF reconstruction networks, we would like to point out a distinction between *inter*-voxel models and *intra*-voxel models. We define inter-voxel models as combining (macro-structural) features *between* voxels in the brain, while intra-voxel models extract (micro-structural) features from *within each* voxel. Prior work in [27] focused primarily on building intra-voxel models. The primary novelty of the architecture we present here is a layer which acts as an inter-voxel model. We expected and have found empirically that combining intra- and inter-voxel models within one network significantly improves performance over using just one of the two. Thus the empirical results presented here should be viewed as complementary to prior work [27] on intra-voxel fODF reconstruction.

Contributions Thus, the main contributions of our work in this paper are: (i) We define the manifold-valued Volterra Series representation for general (complete) Riemannian manifolds and prove that the MVVS is equivariant to translation. Additionally, we prove that the MVC, which is the first-order term of the MVVS, is equivariant to isometry group actions admitted by the manifold. (ii) We present a DNN architecture based on MVVS (MVC), called MVVS(MVC)-Net, for **any** complete Riemannian manifold. (iii) Further, we experimentally demonstrate the performance of the MVVS (MVC)-Net on dMRI classification and fODF reconstruction problems along with comparisons to the state-of-the-art (SOTA). Our results demonstrate significant improvement in accuracy and time efficiency over the SOTA.

The rest of this paper is organized as follows. In section 2, we review background material in Riemannian geometry and the Euclidean Volterra Series. In section 3, we present a novel generalization, the MVVS, of the Volterra Series to manifold-valued functions and prove its equivariance properties. Then, we present a DNN architecture based on MVVS, called the MVVS-Net. In section 4, we present the experimental results and draw conclusions in section 5.

2 Preliminary

In this section, we review some basic material from Riemannian geometry that is necessary in our work and the Volterra Series expansion of nonlinear functions. We

briefly review how the Volterra Series is utilized in the deep learning literature as a higher order alternative to the convolution in CNNs.

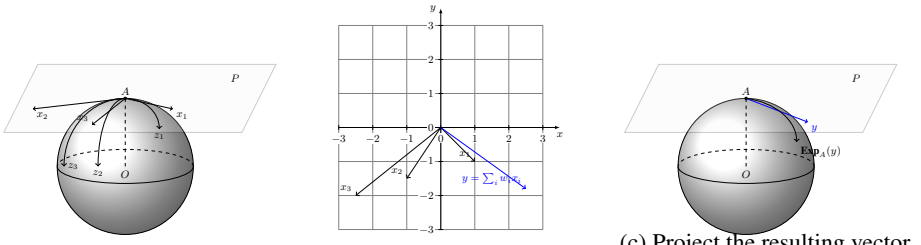
Riemannian Geometry Let (M, g) be a d-dimensional complete Riemannian manifold. The *tangent space* at $p \in M$ is denoted T_pM, which is a d-dimensional vector space. For $p \in M$ and $v \in T_pM$, the geodesic emanating from p with initial direction v is denoted by $\gamma_v(t)$ where $\gamma_v(0) = p$ and $\gamma_v' = v$. The *Exponential map* $\mathbf{Exp}_p : D(p) \subset T_pM \to M$ is defined by $\mathbf{Exp}_p(v) = \gamma_v(1)$ where $D(p) = \{v \in T_pM : \gamma_v(1)$ is defined and $\gamma_v(t)$ is a minimizing geodesic for $0 < t < 1\}$. The exponential map is a diffeomorphism from $D(p)$ to its image, and its inverse is denoted $\mathbf{Log}_p = \mathbf{Exp}_p^{-1}$. These two maps will be of fundamental importance for the construction of the MVVS which will be discussed subsequently.

The Riemannian metric g induces a distance between any two points $p \in M$ and $q \in M$ given by $d_g(p, q) = \inf\{\int_0^1 \sqrt{g(\gamma_{p,q}'(t), \gamma_{p,q}'(t))}dt$: for all $\gamma_{p,q}\}$. Let $x_1, \ldots, x_n \in M$. The Fréchet mean (FM) of $x_1, \ldots, x_n$ is $\bar{x} = \operatorname{argmin}_{m \in M} \sum_{i=1}^n d_g^2(x_i, m)$. This is a generalization of the mean of points in a vector space. For the existence and uniqueness of the FM we refer the reader to [1]. Very briefly, the FM is unique if $x_1, \ldots, x_N$ lie in a open ball of radius r_{cvx}, where r_{cvx} is the *convexity radius* of M [13]. This is often the case in practice and in all our experiments presented subsequently.

For a Riemannian manifold, a metric-preserving diffeomorphism is an isometry. For a smooth map $f : M \to M$, a desired property would be the *isometry equivariance*, i.e. $\phi \circ f = f \circ \phi$ where ϕ is an isometry map. Another similar concept is the *isometry invariance*, i.e. $f \circ \phi = f$.

Volterra Series As is well-known, the traditional convolution is linear shift-invariant. A non-linear shift-invariant system can be approximated by the Volterra Series [26], which is given by $h(x) = \sum_{n=1}^N \int \cdots \int g(\tau_1, \ldots, \tau_n) \prod_{i=1}^n f(x - \tau_i)d\tau_i$, where g is the Volterra kernel. For the case of $N = 1$, $h(x) = \int g(\tau)f(x - \tau)d\tau = (g \star f)(x)$ is the usual convolution.

3 Manifold-Valued Volterra Series and Convolution



(a) Log map all of the data in the window onto the tangent space, i.e. $x_i = \mathbf{Log}_A(z_i)$.

(b) Perform a weighted sum in the tangent space $T_A\mathcal{M}$ to get $y = \sum_i w_i x_i$

(c) Project the resulting vector down using the Riemannian exponential map, i.e. the output is $\mathbf{Exp}_A(y)$.

Fig. 1: Manifold-valued convolution operation within a window.

We now present a novel extension of the Volterra series to manifold-valued functions. We show the first order approximation of the proposed MVVS gives a natural extension of the convolution operation to manifold-valued functions. Further, we show that this

MVC is equivariant to the isometry group action admitted by the manifold and discuss how to use the MVVS/MVC as basic building blocks to design efficient networks for different tasks.

3.1 Manifold-Valued Volterra Series

For manifold-valued data, we can define an analog of the traditional Volterra series. Let $\odot$ be the Hadamard product, i.e. $x_1 \odot x_2 = [x_{11}x_{21}, \ldots, x_{1n}x_{2n}]$ and $\bigodot_{i=1}^{k} x_i = \left[\prod_{i=1}^{k} x_{i1}, \ldots, \prod_{i=1}^{k} x_{in}\right]$. Hadamard product depends on the tangent vector representation and we use the coordinates induced by the **Log** maps, which are given for the sphere and SPD manifold in Section 3.3. Then the MVVS is defined as follows.

Definition 1. *Let (M, g) be a complete Riemannian manifold and $f : Z^d \to M$ be a function defined on Z^d. Let $\{w^{(j)} : (Z^d)^j \to \mathbb{R}\}$ be a collection of kernels. Then*

$$MVVS(f, w^{(1)}, \ldots, w^{(N)})(\mathbf{y}) \coloneqq \\ \boldsymbol{Exp}_{m(\mathbf{y})}\left(\sum_{j=1}^{N} \sum_{z_1,\ldots,z_j} w^{(j)}(z_1 - \mathbf{y}, \ldots, z_j - \mathbf{y}) \bigodot_{i=1}^{j} \boldsymbol{Log}_{m(\mathbf{y})} f(z_i)\right)$$

for $\mathbf{y} \in Z^d$ where $m(\mathbf{y}) = \boldsymbol{FM}(f(\mathbf{z}))$ where $\mathbf{z}$ ranges over the support of the Volterra masks $w^{(j)}$ centered at $\mathbf{y}$.

Note that the FM is computed locally in each window centered at the point $\mathbf{y}$. The most prominent feature of the convolution in Euclidean spaces is translation equivariance (in the domain), which allows weight sharing. Similar to the equivariance to translations (in the domain) of the Volterra series in Euclidean space, the following theorem states that MVVS possesses a similar property.

Theorem 1 (Equivariance to Translation). *Let $h = MVVS(f, w^{(1)}, \ldots, w^{(N)})$, then $h_t = MVVS(f_t, w^{(1)}, \ldots, w^{(N)})$ for all $t \in Z^d$, where, $f_t(z) = f(z - t)$ and $h_t(y) = h(y - t)$.*

The proof follows trivially from the definition of the MVVS through a change of variables and hence we will skip it here. For $N = 1$, we write the MVVS as $MVC(f, w)(y) = \mathbf{Exp}_m\big(\sum_{z \in Z^d} w(z - y)\mathbf{Log}_m f(z)\big)$ which gives us a natural generalization of convolution to manifold-valued functions. An illustration of the MVC operations are depicted in Figure 1. In this work, we also consider the second-order MVVS as a more expressive alternative to the MVC. In the situation with only finite observations at the grid points $z_1, \ldots, z_n \in Z^d$, i.e. we have $x_i = f(z_i)$, $w_i^{(1)} = w^{(1)}(z_i)$, and $w_i^{(2)} = w^{(2)}(z_i)$ for $i = 1, \ldots, n$, we write $MVC(\{x_i\}_{i=1}^n, \{w_i\}_{i=1}^n) = \mathbf{Exp}_m\big(\sum_{i=1}^n w_i \mathbf{Log}_m x_i\big)$ and

$$MVVS(\{x_i\}_{i=1}^n, \{w_i^{(1)}\}_{i=1}^n, \{w_{i,j}^{(2)}\}_{1 \le i,j \le n}) = \\ \mathbf{Exp}_{\bar{x}}\left(\sum_{i=1}^{n} w_i^{(1)} \mathbf{Log}_{\bar{x}} x_i + \sum_{i,j} w_{i,j}^{(2)} \mathbf{Log}_{\bar{x}} x_i \odot \mathbf{Log}_{\bar{x}} x_j\right).$$

The MVC and the MVVS can be used to generalize CNN and its variants to manifold-valued data. Due to the symmetry of the Hadamard product, we can assume $w_{i,j}^{(2)} = 0$ for $i > j$ to reduce the number of parameters.

Besides the translation equivariance in the domain, the Euclidean convolution is also equivariant to the translation in the range. The translation equivariance in the range leads to, for example, the invariance to changes in brightness by a constant. For the case of MVC, the range (of the input function) is however the manifold M and hence the analogous result would be the equivariance to isometry group action admitted by the manifold. The following theorem states that the proposed MVC is equivariant to the isometry group action admitted by the manifold M. From the proof, it is also obvious that this equivariance is not satisfied by the MVVS for $N > 1$.

Theorem 2. *The MVC is equivariant to the isometry group action admitted by M, i.e. $\phi \circ MVC(f, w) = MVC(\phi \circ f, w)$ where $\phi : M \to M$ is an isometry.*

Proof. The proof relies on the fact that the exponential map commutes with the isometry, i.e., $\phi \circ \mathbf{Exp}_p = \mathbf{Exp}_{\phi(p)} \circ d\phi_p$ [19, Prop. 5.9]. Therefore, when the inverse of $\mathbf{Exp}_p$ exists, $\mathbf{Log}_{\phi(p)} = d\phi_p \circ \mathbf{Log}_p \circ \phi^{-1}$. By the invariance of the intrinsic distance metric, the FM is equivariant to the isometry. Since the MVC is a composition of the exponential map, the log map, and the FM, it is equivariant to the isometry group action.

3.2 Manifold-Valued Deep Network Based on MVVS/MVC

The key components of a CNN are the convolutional layers, the non-linear activation function, and the full-connected (FC) layer. To build an analogous manifold-valued deep network, we need equivalent operations in the context of manifold-valued inputs. We propose to replace the convolution by the MVVS. For the non-linear activation function, the most widely used one is ReLU and we suggest a similar operation called the tangent ReLU (tReLU), which is defined as follows. For $x_1, \ldots, x_n \in M$, $\text{tReLU}(x_i) = \mathbf{Exp}_{\bar{x}}(\text{ReLU}(\mathbf{Log}_{\bar{x}}(x_i)))$ where $\bar{x}$ is the FM of $x_1, \ldots, x_n$ and $\text{ReLU}(x) = \max(x, 0)$ is applied component-wise to its argument. Note that a similar operation was proposed by [9] but they restricted it to the hyperbolic spaces while ours is valid for general complete Riemannian manifolds. Finally, to design a deep network that is invariant to isometry group actions, we need the FC layers to be invariant to the isometry (since the MVC layers are equivariant to the isometry by Theorem 2). In this work, we consider the invariant FC layer proposed in [8] which is constructed by first transforming x_i to $d_i = d_g(x_i, \bar{x})$ and then feeding the d_i's to the usual FC layers. Replacing the MVC with the MVVS, we have a higher order manifold-valued deep network.

Another concern is the extra parameters, i.e. the weights, required by the MVVS compared to the MVC. Note that for a fixed filter size d the number of weights in the MVC is d^2 and for the second-order MVVS is $d^2 + d^2(d^2 + 1)/2$ which is a substantial increase. A way to mitigate this problem is to assume that the kernel $w^{(2)}(z_1, z_2)$ is *separable*, that is, $w^{(2)}(z_1, z_2) = w_1(z_1)w_2(z_2)$. Under this assumption, the number of weights is $3d^2$, which is in the same order as the MVC. The separability of the kernel is assumed in all of our experiments.

We like to emphasize that the proposed MVC/MVVS and the tReLU operations are substitutions for the Euclidean space convolution and the ReLU operations respectively. In the next section, we present closed form Riemannian Exp and Log operations for the manifolds we use in the experiments.

3.3 The cases of S^n and SPD(n)

Here we specify concrete versions of the building blocks (Exp and Log maps) presented above for particular application domains in dMRI processing. We will tackle two fundamental problems in dMRI processing using this framework: 1) diffusion tensor imaging classification and 2) fODF reconstruction from severely undersampled data.

Diffusion Tensor Image Classification Diffusion tensor imaging (DTI) is a simple and popular model in dMRI processing. Diffusion tensors (DTs) are 3×3 SPD matrices [4]. A dMRI scan processed using the DTI model will output a 3D field of DTs $f : \mathbb{Z}^3 \rightarrow$ SPD(3). Closed form expressions of the Riemannian **Log** and **Exp** maps for the SPD(3) manifold with the GL(n)-invariant metric are given by

$$\mathbf{Exp}_Y(X) = Y^{1/2} \exp(Y^{-1/2} X Y^{-1/2}) Y^{1/2} \text{ and } \mathbf{Log}_Y(X) = Y^{1/2} \log(Y^{-1/2} X Y^{-1/2}) Y^{1/2}$$

where $\exp, \log$ are the matrix exponential and logarithmic maps, respectively.

fODF reconstruction Accurate reconstruction of the fODF from undersampled $S(k, q)$ data has the potential to significantly accelerate dMRI acquisition. Here we present a framework for achieving this. Our fODF reconstruction method performs convolutions on the unit hypersphere S^n. The closed form expressions for the **Log** and **Exp** maps on the sphere are given by the following expression, where $U = X - \langle X, Y \rangle Y$ [28].

$$\mathbf{Exp}_Y(X) = \cos(\|X\|)Y + \sin(\|X\|)\frac{X}{\|X\|} \text{ and } \mathbf{Log}_Y(X) = U \cos^{-1}(\langle X, Y \rangle)/\langle U, U \rangle$$

4 Experiments

In this section we present several real data experiments demonstrating the performances of MVC-net and MVVS-net respectively.

4.1 Parkinson's Disease vs. Controls Classification

We now present an application of the MVC-Net and the MVVS-Net to the problem of classification of Parkinson's disease (PD) patients vs controls. The dataset consists of dMRI scans acquired from 355 PD patients and 356 controls. The acquisition parameters were, # of gradient directions =64, $b = 0, 1000 s/mm2$, repetition time = 7748 ms, echo time= 86 ms, field of view = $(224, 224)$ mm, slice thickness of 2mm, matrix size of $(112, 112)$.

Model	Non-linearity	# params.	time (s) / sample	Accuracy	
				Test Accuracy (60/40)	Test Accuracy (90/10)
MVVS-net	tReLU	$\sim 23K$	~ 0.34	**0.966**	**0.973**
MVC-net	tReLU	$\sim$ **14K**	~ 0.13	0.942	**0.973**
DTI-ManifoldNet [7]	None	$\sim 30K$	~ 0.3	0.934	0.948
ODF-ManifoldNet [7]	tReLU	$\sim 153K$	~ 0.02	0.941	0.942
ResNet-34 [14]	ReLU	$\sim 30M$	$\sim$ **0.008**	0.708	0.713
CapsuleNet [25]	ReLU	$\sim 30M$	~ 0.009	0.618	0.622

Table 1: Comparison results for PD vs. Controls classification.

From each of these dMRIs, 12 regions of interest (ROIs) – six on each hemisphere of the brain – in the sensorimotor tract are segmented by registering to the sensorimotor area tract template (SMATT) [2]. These tracts are known to be affected by PD. For this experiment, we adopt the most widely used representation of dMRI in the clinic namely, the DTI and also to demonstrate that our methods work well for the SPD manifold. DTs are 3×3 SPD matrices [4]. Each of the ROIs (12 in total) contain 26 voxels. For each patient (control), all the ROIs are concatenated together to form a $12 \times 26 \times 3 \times 3$ input tensor to the network. The output is a binary class label specifying whether the input image came from a PD or control.

Architecture The MVC-Net architecture is obtained from the traditional CNN by replacing the convolution operations with MVC (and MVVS) operations and the ReLU with **tReLU**. For this experiment, the MVC-net consists of five **MVC** + **tReLU** layers. Each of the MVC (MVVS) layers has a window size of 4 and a stride of 1. We use the closed form exponential and log maps for the SPD(n) manifold presented in 3.3.

Experimental Results In this experiment on PD vs. Control classification from DTI brain scans, we compared the performance of MVC-Net and MVVS-Net with several deep net architectures including the ManifoldNet [7, 8] the ResNet-34 architecture [14] and a CapsuleNet architecture [25] with dynamic routing. To perform the comparison, we applied each of the aforementioned deep net architectures to the above described diffusion tensor image data sets. For the ResNET-34 and CapsuleNet, we vectorize the diffusion tensors as these networks are applicable only to vector space data.

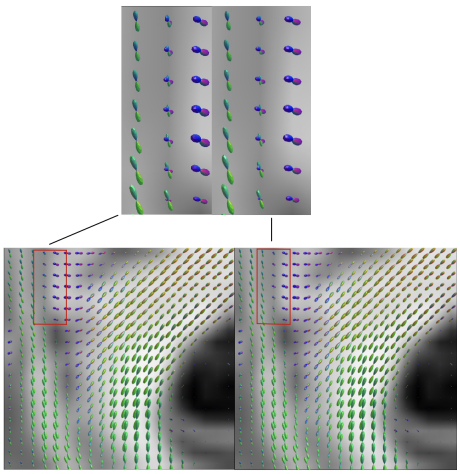


Fig. 2: Left: HCP sample patch from centrum semiovale ground truth/gold standard fODF. Right: Network reconstruction from 7% sampled data. Zoomed-in figures display a particularly hard crossing-fiber ROI.

We train our MVC-net architecture for 200 epochs using the cross-entropy loss and an Adam optimizer with the learning rate set to 0.005. We report two different results for each architecture. One is obtained on a $90/10$ training to test split. Since the results for the top performing architectures in this category were all high, we also report a more challenging $60/40$ training to test split to obtain more differentiation between the methods.

As is evident from the Table 1, MVC-net and MVVS-Net outperform all other methods on both training and test accuracy while simultaneously keeping the lowest parameter count. MVVS-net either is equal ($90/10$ split) or outperforms ($60/40$ split) MVC-net, as expected from the increased model capacity of the MVVS. The inference speeds under-perform ResNet-34 and CapsuleNet, but these architectures utilize operations that were optimized heavily for inference speed over the years. Further, in terms of the possible application domain of automated PD diagnosis, the inference speeds we have achieved are more than sufficient in practice.

4.2 fODF Reconstruction

In this experiment, we consider the problem of reconstructing fODFs from compressed sensed (CS) dMRI data. Specifically, given sub-Nyquist sampled (compressed sensed in the 6-dimensional $(\mathbf{k}, \mathbf{q})$ Fourier space) dMRI data, we seek to reconstruct a field of fODF that characterize the diffusional properties of tissue being imaged. The goal of the network will be to learn the highly non-linear mapping between an under-sampled

(aliased) reconstruction of the fODF field to the fully-sampled reconstruction of the fODF field.

The fODF can be obtained from fully sampled data using a constrained spherical deconvolution [29]. The fODF is a real-valued positive function on the sphere $f : \mathbb{S}^2 \to \mathbb{R}^+$ and after normalization can be represented as a square-root density, i.e. a point on the unit Hilbert sphere. For sampled fODFs, this representation reduces to a point on the unit hypersphere, $\mathbb{S}^{n-1}$. This unit hypersphere representation will be used in the inter-voxel layers, while the sampled $f : \mathbb{S}^2 \to \mathbb{R}^+$ representation will be used in the intra-voxel layers, leveraging a recent architecture introduced in [10]. that will be elaborated on below. For the inter-voxel layers, we will use MVC and MVVS convolution layers on the unit hypersphere manifold, with closed form expressions for the **Exp** and **Log** maps respectively as presented in 3.3.

Data Description We test our fODF reconstruction network on real data from the Human Connectome Project (HCP) [31]. Since the HCP data is acquired with extremely dense sampling, we consider the fODF reconstructions from these HCP scans as the ground truth/gold standard. fODFs in this case are generated using MSMT-CSD [17]. implemented in the *mrtrix3* library [30] which guarantees positivity of the fODF amplitudes. fODFs are represented by sampling on a consistent spherical grid consisting of 768 points in the Healpix sampling [12].

For under-sampling, we apply an inverse power-law under-sampling scheme (see [22]) in the $(\mathbf{k}, \mathbf{q})$ space, which is the data acquisition space.

The training data sets consist of pairs of aliased (under-sampled) and ground-truth (fully sampled) fODF field reconstructions. The goal of the network is to learn to reconstruct the fully sampled fODF field from the input aliased fODF field reconstruction. Due to limited computational resources, in this experiment, we only consider patches of size 21×21 in a slice, i.e., one training sample is a pair consisting of an under-sampled 21×21 patch reconstruction and a fully sampled reconstruction of the same patch. This patch-based approach is quite common in CS-based reconstruction algorithms.

For the real data, we extract the 21×21 voxel ROI from a large subset of HCP scans (432 in total) in the centrum semiovale where projection, commissural and association tracts cross and pose a great challenge for under-sampled reconstruction. We use 40 random samples for testing and train on the remaining 392 samples.

Architecture As explained previously, the network consists of two components: an *intra-voxel* component which operates individually inside each voxel and an *inter-voxel* component which combines features across voxels. The inter-voxel component consists of a series of **MVC** $\to$ **tReLU** or **MVVS** $\to$ **tReLU** blocks. The input to these blocks is a $H \times W \times C \times N$ tensor representing a patch within a slice of the dMRI scan,

Method	HCP Data					
	MAE (7 %)	MAE (11 %)	MAE (20 %)	bNMSE (7 %)	bNMSE (11 %)	bNMSE (20 %)
MVVS + SphereConv	**9.31**	**9.29**	**7.41**	**0.24**	**0.40**	**0.38**
MVC + SphereConv	10.12	9.43	7.42	0.28	0.41	0.43
SphereConv [10]	13.92	12.61	10.76	0.34	0.64	0.65
$\mathbb{S}^2$ **U-net** [27]	11.04	10.93	8.03	0.31	0.57	0.59
3D CNN [20]	11.88	11.60	8.77	0.35	0.61	0.65
MSMT-CSD (baseline)	16.81	16.32	12.14	1	1	1

Fig. 3: Comparison results on dMRI fODF reconstruction. The number in parenthesis indicates the sampling rate of the under-sampled reconstruction input.

where N is the number of sample points of the fODF spherical function and C the number of channels. For example, in the real data experiments, we have an initial input size of $21 \times 21 \times 1 \times 768$. The intra-voxel model needs to process the data within voxels, i.e., the individual fODFs. We design and implement a novel intra-voxel layer using a spherical convolution layer that we denote by **SphereConv** presented in the recent DeepSphere paper [10]. This layer represent the spherical signal of the fODF as a graph with node weights equal to the fODF value at the sample points, and applies spectral graph convolutions to transform the signal. There are approximate rotational equivariance guarantees for **SphereConv** that fit the fODF reconstruction problem well. We would like to stress that the choice of intra-voxel layer is orthogonal to the novelty of this work, namely the inter-voxel **MVC** and **MVVS** convolutions.

In summary, the inter-voxel component combines features between voxels by using the **MVC** layer, while the intra-voxel component shares weights between all voxels but has the capacity to learn within the voxel. We found that applying the inter-voxel layers first, followed by intra-voxel layers later gives optimal performance. With these details in mind, we used the following architecture for real data fODF reconstruction.

$$\textbf{MVVS}(1,8) \rightarrow \textbf{MVVS}(8,16) \rightarrow \textbf{MVVS}(16,32) \rightarrow \textbf{MVVS}(32,32) \rightarrow 7\times(\textbf{SphereConv})$$

where, $\textbf{MVVS}(C_i, C_o)$ represents an MVVS layer with C_i input and C_o output channels respectively. All layers use a kernel size of 3 and a stride of 1. The **SphereConv** layers have feature channels $32 \rightarrow 64 \rightarrow 128 \rightarrow 256 \rightarrow 256 \rightarrow 128 \rightarrow 48 \rightarrow 1$ and use a U-net style architecture, i.e., with channel concatenation between encoder and decoder layers. All **MVVS** and **SphereConv** layers are followed by a **tReLU** and **ReLU** operation respectively. Results for the same architecture but using **MVC** layers instead of **MVVS** are also presented. For training, the Adam optimizer with an initial learning rate of 0.03 is used. We use an MSE function *weighted by the fractional anisotropy of the undersampled ground truth image* as the reconstruction loss function during training. This FA-weighted MSE encourages the network to focus more on reconstruction of highly anisotropic voxels which in some cases was found to improve visual results substantially. It is possible that this loss could give low weight to crossing fiber voxels (which will appear as low FA regions), but no visual degradation was observed in these regions.

Experimental Results We quantitatively measure the model performance using mean-angular error (MAE) and baseline normalized MSE (bNMSE). The MAE is computed for only crossing fiber voxels using the method presented in the experiments of [27]. In summary, a threshold of 0.1 of the largest peak is used to eliminate spurious fibers, and all corresponding two-peak voxels from the network output and ground truth are compared using the angular error in degrees. The bNMSE is defined as $\text{MSE}(F_g, F_o)/\text{MSE}(F_g, F_i)$, where F_g, F_o and F_i are the ground truth fODF, the network output and the undersampled (aliased) fODF respectively. Thus the bNMSE compares the accuracy of the network output to the accuracy of the baseline method (MSMT-CSD in this case), where lower values indicate more improvement relative to the baseline method. This metric was used in place of $\text{MSE}(F_g, F_o)$ to allow more robust comparisons with competing methods, given that results reported in competing methods were most likely obtained

from different ROIs and hence difficult to compare to without knowing the precise ROI localization, thus a direct MSE comparison may bias results.

All models are trained for 1000 epochs on a single Quadro RTX 6000 GPU (about 64 hours total training time). Table 3 reports the results for HCP data experiments. As evident, for all sampling rates, our method outperforms other deep learning and the baseline (MSMT-CSD) methods in terms of both MAE and bNMSE. Visualization results shown in figure 2 are similarly compelling. The zoomed in area shows a difficult crossing fiber pattern which the network has reconstructed quite well. These results constitute improvements that can reduce dMRI scan acquisition time by orders of magnitude while retaining image quality. Moreover, from an ablation view point, we see that the **MVC** layers (the inter-voxel component) improves accuracy substantially over just the intra-voxel **SphereConv** layers, and **MVVS** further improves the accuracy. Note that our chosen intra-voxel layer actually performs worse in all cases than the intra-voxel layer presented in [27]. This suggests that further improvements could be made by combining our novel inter-voxel **MVVS/MVS** layers with [27] which will be explored in our future work.

5 Conclusion

In this paper, we presented a novel higher order CNN for manifold-valued images. We defined the the analog of the traditional convolutions for manifold-valued images and proved powerful equivariance properties. Finally, we presented experiments demonstrating the superior performance of the MVC (MVVS)-Net in comparison to other SOTA methods on important problems in dMRI.

References

1. Afsari, B.: Riemannian L^p center of mass: Existence, uniqueness, and convexity. Proc. of the AMS **139**(02), 655–655 (2011). https://doi.org/10.1090/S0002-9939-2010-10541-5
2. Archer, D., Vaillancourt, D., Coombes, S.: A template and probabilistic atlas of the human sensorimotor tracts using diffusion MRI. Cerebral Cortex **28**, 1–15 (03 2017). https://doi.org/10.1093/cercor/bhx066
3. Banerjee, M., Chakraborty, R., Bouza, J., Vemuri, B.C.: Volterranet: A higher order convolutional network with group equivariance for homogeneous manifolds. IEEE Trans. on PAMI (01), 1–1 (nov 5555). https://doi.org/10.1109/TPAMI.2020.3035130
4. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophysical journal **66**(1), 259–267 (1994)
5. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P.: Geometric deep learning: going beyond Euclidean data. IEEE Signal Processing Magazine **34**(4), 18–42 (2017)
6. Brooks, D., Schwander, O., Barbaresco, F., Schneider, J.Y., Cord, M.: Riemannian batch normalization for SPD neural networks. In: Advances in NeurIPS. pp. 15463–15474 (2019)
7. Chakraborty, R., Bouza, J., Manton, J., Vemuri, B.C.: Manifoldnet: A deep neural network for manifold-valued data with applications. IEEE Trans. on PAMI pp. 1–1 (2020)
8. Chakraborty, R., Bouza, J., Manton, J., Vemuri, B.C.: A deep neural network for manifold-valued data with applications to neuroimaging. In: Intl. Conf. on IPMI. pp. 112–124. Springer (2019)
9. Chami, I., Ying, Z., Ré, C., Leskovec, J.: Hyperbolic graph convolutional neural networks. In: Advances in NeurIPS. pp. 4869–4880 (2019)
10. Defferrard, M., Milani, M., Gusset, F., Perraudin, N.: Deepsphere: a graph-based spherical cnn. In: ICLR (2019)

11. Dell'Acqua, F., JD, T.: Modelling white matter with spherical deconvolution: How and why? NMR in Biomedicine **32**, e3945 .https://doi.org/10.1002/nbm.394518 (2017)
12. Gorski, K.M., Hivon, E., Banday, A.J., Wandelt, B.D., Hansen, F.K., Reinecke, M., Bartelmann, M.: Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. The Astrophysical Journal **622**(2), 759 (2005)
13. Groisser, D.: Newton's method, zeroes of vector fields, and the Riemannian center of mass. Advances in Applied Mathematics **33**(1), 95–135 (2004). https://doi.org/10.1016/j.aam.2003.08.003
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE CVPR. pp. 770–778 (2016)
15. Huang, Z., Van Gool, L.J.: A Riemannian Network for SPD Matrix Learning. In: AAAI. vol. 1, p. 3 (2017)
16. Huang, Z., Wu, J., Van Gool, L.: Building deep networks on Grassmann manifolds. In: 32 AAAI Conf. on Artificial Intelligence (2018)
17. Jeurissen, B., Tournier, J.D., Dhollander, T., Connelly, A., Sijbers, J.: Multi-tissue constrained spherical deconvolution for improved analysis of multi-shell diffusion MRI data. Neuroimage **103**, 411–426 (2014)
18. Kumar, R., Banerjee, A., Vemuri, B.C., Pfister, H.: Trainable convolution filters and their application to face recognition. IEEE Trans. on PAMI **34**(7), 1423–1436 (2011)
19. Lee, J.M.: Riemannian manifolds: an introduction to curvature, vol. 176. Springer Science & Business Media (2006)
20. Lin, Z., Gong, T., Wang, K., Li, Z., He, H., Tong, Q., Yu, F., Zhong, J.: Fast learning of fiber orientation distribution function for mr tractography using convolutional neural network. Medical physics **46**(7), 3101–3116 (2019)
21. Lucena, O., Vos, S.B., Vakharia, V., Duncan, J., Ourselin, S., Sparks, R.: Convolutional neural networks for fiber orientation distribution enhancement to improve single-shell diffusion mri tractography. In: Computational Diffusion MRI, pp. 101–112. Springer (2020)
22. Lustig, M., Donoho, D., Pauly, J.: Sparse MRI: The application of compressed sensing for rapid MR imaging. MRM **58**(6), 1182–1195 (2007)
23. Maurice Fréchet: Les éléments aléatoires de nature quelconque dans un espace distancié. Annales de l'I. H. P., **10**(4), 215–310 (1948)
24. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. IJCV **66**(1), 41–66 (2006)
25. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Advances in NeurIPS. pp. 3856–3866 (2017)
26. Schetzen, M.: The Volterra and Wiener Theories of Nonlinear Systems. Wiley (1980)
27. Sedlar, S., Papadopoulo, T., Deriche, R., Deslauriers-Gauthier, S.: Diffusion MRI fiber orientation distribution function estimation using voxel-wise spherical U-net. In: Computational Diffusion MRI, MICCAI Workshop (2020)
28. Srivastava, A., Jermyn, I., Joshi, S.: Riemannian analysis of probability density functions with applications in vision. In: 2007 IEEE CVPR. pp. 1–8. IEEE (2007)
29. Tournier, J.D., Calamante, F., Connelly, A.: Robust determination of the fibre orientation distribution in diffusion MRI: non-negativity constrained super-resolved spherical deconvolution. Neuroimage **35**(4), 1459–1472 (2007)
30. Tournier, J.D., Smith, R., Raffelt, D., Tabbara, R., Dhollander, T., Pietsch, M., Christiaens, D., Jeurissen, B., Yeh, C.H., Connelly, A.: Mrtrix3: A fast, flexible and open software framework for medical image processing and visualisation. Neuroimage **202**, 116137 (2019)
31. Van Essen, D.C., Smith, S.M., Barch, D.M., Behrens, T.E.J., Yacoub, E., Ugurbil, K., Consortium, W.M.H.C.P., Others: The WU-Minn human connectome project: an overview. Neuroimage **80**, 62–79 (2013)