





# Geometric Deep Learning for Unsupervised Registration of Diffusion Magnetic Resonance Images

Jose J. Bouza<sup>1</sup>, Chun-Hao Yang<sup>2</sup> , and Baba C. Vemuri<sup>3</sup> 

<sup>1</sup> Intuitive Surgical, 1020 Kifer Road, Sunnyvale, CA, USA

<sup>2</sup> Institute of Applied Mathematical Sciences, National Taiwan University, Taipei, Taiwan  
chunhaoy@ntu.edu.tw

<sup>3</sup> Department of CISE, University of Florida, Gainesville, FL, USA  
vemuri@ufl.edu

**Abstract.** Deep learning based models for registration predict a transformation directly from moving and fixed image appearances. These models have revolutionized the field of medical image registration, achieving accuracy on-par with classical registration methods at a fraction of the computation time. Unfortunately, most deep learning based registration methods have focused on scalar imaging modalities such as T1/T2 MRI and CT, with less attention given to more complex modalities such as diffusion MRI. In this paper, to the best of our knowledge, we present the first end-to-end geometric deep learning based model for the non-rigid registration of fiber orientation distribution fields (fODF) derived from diffusion MRI (dMRI). Our method can be trained in a fully-unsupervised fashion using only input fODF image pairs, i.e. without ground truth deformation fields. Our model introduces several novel differentiable layers for local Jacobian estimation and reorientation that can be seamlessly integrated into the recently introduced manifold-valued convolutional network in literature. The results of this work are accurate deformable registration algorithms for dMRI data that can execute in the order of seconds, as opposed to dozens of minutes to hours consumed by their classical counterparts.

**Keywords:** Registration · Geometric Deep Learning · Diffusion MRI

## 1 Introduction

Image registration is a fundamental operation in medical image analysis. Broadly speaking, the image registration problem involves finding the correspondence (match) between images in different coordinate systems. This correspondence can be established by finding an appropriate geometric transformation between the coordinate systems. The nature of this transformation can vary from a simple global affine transformation to a full non-rigid transformation yielding a dense deformation field. In this paper, we develop the first end-to-end deep learning based model for full-deformable registration of fiber orientation distribution function (fODF) fields derived from diffusion MRI (dMRI) data. Our method can be easily ported (after some simple modifications) to other commonly used derived representations from dMRI such as, the diffusion tensor (DT) or the ensemble average propagator (EAP) fields.

This paper firstly builds in novel ways on the existing manifold valued convolution operations presented in [5] and then develops a novel architecture suitable for the non-rigid registration of fODFs derived from dMRI. Our novel contributions include: 1) An efficient CUDA implementation of the core operations presented in [5] 2) Several differentiable layers required for diffusion MRI registration (Jacobian estimation layer, reorientation layer) 3) Design and implementation of end-to-end deformable dMRI registration networks 4) A detailed experimental analysis of these models.

### 1.1 Prior Work: Classical dMRI Registration

Image registration is a fundamental problem in medical image analysis with numerous applications. We now present a brief note on dMRI registration applied to derived representations such as: diffusion tensor images (DTI), ensemble average propagator (EAP) fields or fiber orientation density function (fODF) fields. All of these images are manifold-valued images in that, at each voxel, we have a manifold-valued 'object'. In the case of DTI, this object is an element of the manifold of  $(n, n)$  symmetric positive definite matrices denoted by  $P_n$ , for EAP (fODF) fields/images, it is the manifold of probability density functions.

Early work on DTI registration used a registration cost function based on either fractional anisotropy (FA) or some rotation invariant features computed from the DTs [14, 26] These methods are however not applicable to higher order tensor field representation which might be needed to cope with crossing fibers in the data. In this context, groupwise registration of fourth order tensor representations of dMRI data was presented in [3]. In [29], a non-rigid registration and reorientation algorithm applied directly to the raw dMRI data was presented. Their algorithm performs the reorientation via the use of pre-specified fiber basis functions.

Several approaches to register the EAP (fODF) fields have been proposed in literature. These methods first compute EAPs (fODFs) at each voxel and then register these derived manifold-valued fields/images [7, 17]. For an extensive literature review, we refer the reader to [10].

### 1.2 Prior Work: Deep Learning Based Registration

Modern classical registration algorithms are relatively accurate, but require substantial computation time due to their iterative nature. It is not uncommon for even well-optimized software such as ANTs [1] to take upwards of 30 min to register a pair of high-resolution brain MRI volumes. With the introduction of deep learning based registration methods the possibility of registration with accuracy on-par with classical methods, but with a runtime on the order of seconds is within reach.

The first deep learning based registration methods [27] required ground truth deformation fields and processed image patches instead of full images. Later methods used fully unsupervised methods for training but still using a path-based approach [16, 23]/With the introduction of the VoxelMorph architecture [2], Balakrishnan et al. showed that full-resolution image registration is possible within a deep network. Further work has extended the VoxelMorph architecture to guarantee diffeomorphic registrations [8] and learning contrast-invariant registrations [11]. These recent models achieve

accuracy on-par with classical registration algorithms, with several orders of magnitude improvements in runtime. All the above methods are however designed for registration of scalar-valued images.

In this paper, we focus on the task of dMRI registration. Registration of dMRI data is more challenging than traditional modalities for several reasons. First, because dMRI contains directional information, a reorientation step must follow the application of a deformation field. Second, dMRI data has substantially more information at each voxel than most other modalities, and thus requires more memory and computation to process. Finally, in the context of deep learning based methods, there has been little attention given to developing network architectures that respect the manifold geometry of the dMRI derived image representations such as DTI, fODF images etc., which form the input to the network or can be estimated within the network.

To the best of our knowledge, the only existing deep-learning based dMRI registration method in literature is DDMReg [28]. This model extracts FA images and several tract orientation maps (TOMs) from the dMRI data. Each FA and TOM image is passed through a separate registration subnetwork (each of which is a VoxelMorph style architecture [2]). Each subnetwork outputs a proposal deformation field, and a multi-deformation fusion subnetwork combines these fields to generate a final predicted deformation field. This approach has a few pitfalls. First, the registration subnetworks and fusion subnetworks are all trained separately, thus not achieving the performance of end-to-end trained models. Second, the FA and TOM inputs are hand-crafted features extracted from the dMRI data which consumes preprocessing time during inference. Further, we show that we can achieve improved performance by building a model that can directly process dMRI derived (fODF) data in a way that respects the underlying geometry.

### 1.3 Paper Organization

In Sect. 2 we briefly review the Manifold Valued Convolution (MVC) and Manifold Valued Volterra Series (MVVS) operations, the core layers used to process the fODF fields derived from dMRI. In Sect. 3, we present efficient CUDA implementations of the MVC and MVVS operations. Section 4 contains a description of our deep network architectures for deformable registration, including a differentiable implementation of a dMRI reorientation method. Finally, Sect. 5 contains an extensive set of experimental results demonstrating the performance of our geometric deep network.

## 2 Manifold Valued Volterra Series

We will now very briefly review the manifold valued convolution (MVC) and manifold valued Volterra series (MVVS) operations introduced in [5]. A manifold-valued image is a map  $F : \mathbb{Z}^n \rightarrow \mathcal{M}$  and this image modality naturally arises in various dMRI data representations. Recall that for  $x_1, x_2 \in \mathbb{R}^n$ , the Hadamard product of  $x_1$  and  $x_2$  is  $x_1 \odot x_2 = [x_{11}x_{21}, \dots, x_{1n}x_{2n}]$ . The  $N^{th}$  order Volterra series for  $g : \mathbb{R}^n \rightarrow \mathbb{R}$  and  $f : \mathbb{R} \rightarrow \mathbb{R}$  is given by  $h(x) = \sum_{n=1}^N \int \dots \int g_n(x - \tau_1, \dots, x - \tau_n) \prod_{i=1}^n f(\tau_i) d\tau_i$ .

In this work, we only consider the first and second order MVVS, which are defined by

$$\begin{aligned}
MVC(F, w)(\mathbf{y}) &= \mathbf{Exp}_{m(\mathbf{y})} \left( \sum_{\mathbf{z}=1}^K w(\mathbf{z} - \mathbf{y}) \mathbf{Log}_{m(\mathbf{y})} F(\mathbf{z}) \right) \\
MVVS(F, w_1, w_2)(\mathbf{y}) &= \mathbf{Exp}_{m(\mathbf{y})} \left[ \sum_{\mathbf{z}=1}^K w_1(\mathbf{z} - \mathbf{y}) \mathbf{Log}_{m(\mathbf{y})} F(\mathbf{z}) \right. \\
&\quad \left. + \sum_{\mathbf{z}_1, \mathbf{z}_2=1}^K w_2(\mathbf{z}_1 - \mathbf{y}, \mathbf{z}_2 - \mathbf{y}) \left( \mathbf{Log}_{m(\mathbf{y})} F(\mathbf{z}_1) \right) \odot \left( \mathbf{Log}_{m(\mathbf{y})} F(\mathbf{z}_2) \right) \right]
\end{aligned}$$

where  $F : \mathbb{Z}^d \rightarrow \mathcal{M}$  is a manifold-valued image and  $w_j : (\mathbb{Z}^d)^j \rightarrow \mathbb{R}$  is the  $j$ -th kernel with size  $K$ . For  $\mathbf{y} \in \mathbb{Z}^d$  where  $m(\mathbf{y}) = \mathbf{FM}(F(\mathbf{z}))$  where each  $\mathbf{z}$  ranges over the support of the Volterra masks  $w_j$  centered at  $\mathbf{y}$  and  $\mathbf{FM}$  is the unweighted Frechet mean. The  $N$ -th order MVVS is a straightforward generalization and we refer the reader to [5].

In [4], it was reported that using the sample in the middle voxel of the moving window gives similar performance to using the Frechet mean. This modification significantly improves performance and ease of implementation, thus we opt to use the mid point of the moving window as the base point throughout this paper. In the interest of computational and parameter efficiency, we only use MVC and second-order MVVS layers in our experiments.

### 3 Implementation

In this section, we present a novel CUDA implementation of the MVVS layer. This implementation allows us to build networks for processing the fODF field derived from the full dMRI volume.

#### 3.1 Data of Interest

In this work we limit our focus to the fODF representation of dMRI data [18]. An fODF describes the distribution of fiber orientations in a voxel. The diffusion signal is modeled as the convolution of the fODF and the response function characterizing diffusion along coherent fiber bundles. Thus, given the diffusion signal and the response function, the fODF can be solved for via deconvolution [20]. The space of all fODFs can be defined by the set  $\Phi$ :  $\left\{ \phi : \hat{\mathbf{u}} \in S^2 \rightarrow R_0^+ \mid \phi(\hat{\mathbf{u}}) \geq 0, \int_{S^2} \phi(\hat{\mathbf{u}}) d\hat{\mathbf{u}} = 1 \right\}$  where,  $R_0^+$  is the set of nonnegative reals. Using a square root density parameterization, this distribution can be identified with a point on the unit Hilbert sphere, a Riemannian manifold whose geometry is fully known, and has been used in literature for EAP (fODF) estimation [22]. We represent a sampled fODF as a point on the unit hypersphere  $\mathbb{S}^M$  following the convention in [5], where  $M$  is the number of sample points. The unit hypersphere is a Riemannian manifold, thus this representation fits well into the MVC/MVVS framework. By virtue of the fact that this representation of fODF leads to elements in the space of probability distributions, unlike the spherical harmonic representation of fODFs, it does not require explicit enforcement of non-negativity and integration to one constraints.

### 3.2 CUDA Implementation

We now present an optimized CUDA implementation of the MVC operation. This allows us for the first time to use these operations at the scale of full dMRI volumes, unlike the original MVC work presented in [5]. For clarity, we first present a naive CUDA implementation and then briefly describe several optimizations added to the naive implementation. We only present the MVC operation details in this section, but we also implemented forward and backward passes for the MVVS operation. The code is made public.

The input manifold-valued image will be represented by a tensor of shape  $C \times D \times W \times H \times M$ , where  $C, D, W, H$  are the channels, depth, width and height respectively. The output manifold-valued image will be a tensor of shape  $C_{\text{out}} \times S(D) \times S(W) \times S(H) \times M$  where  $S(x) = \frac{x-K}{T} + 1$ ,  $K$  is the filter size and  $T$  the stride. The data at each voxel is a point on the hypersphere embedded in Euclidean space,  $S^{M-1} \subset \mathbb{R}^M$ , and is thus an  $M$  dimensional vector. The weight filter will be represented as a tensor of shape  $C \times C_{\text{out}} \times K^3$  where  $C_{\text{out}}$  is the number of output channels and  $K$  is the filter size.

In the naive CUDA implementation, each CUDA thread will compute one output voxel. Suppose a thread is assigned to compute voxel  $\mathbf{v} = [c, d, w, h]$  in the output image. It will perform the following steps:

1. Compute the input voxels coordinates in the receptive field of the output voxel  $\mathbf{v}$ . Let  $R_i(\mathbf{v})$  denote the  $i$ th voxel in the receptive field of  $\mathbf{v}$ .
2. Compute the Riemannian Log of each voxel value in the receptive field with base point set to the midpoint of the receptive field.  $\mathbf{Log}_{R_m(\mathbf{v})}(R_i(\mathbf{v}))$  for  $i = 1, \dots, K^3$  where  $m$  is the index of the midpoint.
3. Perform the following weighted sum,  $T = \sum_{i=1}^{K^3} w_i \mathbf{Log}_{R_m(\mathbf{v})}(R_i(\mathbf{v}))$  where  $w_i$  are the weight filter values.
4. Perform the Riemannian exponential  $\mathbf{Exp}_{R_m(\mathbf{v})}(T)$  and write it to the output manifold-valued image at voxel coordinate  $\mathbf{v}$ .

The closed form expressions for the **Log** and **Exp** maps on the sphere are given by the following expression, where  $U = X - \langle X, Y \rangle Y$  [19].  $\mathbf{Exp}_Y(X) = \cos(\|X\|)Y + \sin(\|X\|) \frac{X}{\|X\|}$  and,  $\mathbf{Log}_Y(X) = U \cos^{-1}(\langle X, Y \rangle) / \langle U, U \rangle$ .

We optimize the naive CUDA implementation with two strategies: 1) remove temporary memory allocations by performing the Riemannian Log, weighted sum and Riemannian Exp in-place (steps 2–4). 2) Use tiling [25] to reduce redundant global memory reads by using shared memory to do spatial caching.

**Performance Analysis.** We perform a benchmark analysis of the following implementations of the MVC operation: 1. PyTorch CPU Implementation 2. PyTorch GPU Implementation 3. Naive CUDA Implementation 4. Memory Optimized CUDA Implementation 5. Tiled CUDA Implementation. The metrics of interest are runtime and peak global GPU memory usage. All experiments are run on an RTX 2080 Ti GPU with CUDA version 11.1. A random  $\mathbb{S}^{M-1}$ -valued image was generated with spatial dimensions  $80^3$  and  $M = 45$ . A random weight kernel was generated with kernel size  $K = 3$ .

The input and output channels are both 1. A CUDA thread block size of  $5^3$  is used. All reported metrics are averaged over 10 runs.

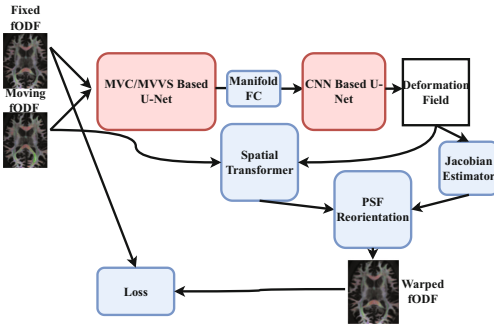
**Table 1.** Performance of tested MVC Implementations. Memory usage is measured as peak GPU global memory usage.

Method	Runtime (s)	Memory Usage (MB)
PyTorch CPU	15.646	N/A
PyTorch GPU	0.378	9627
Naive CUDA	0.162	452
Memory-Efficient (M-E) CUDA	0.154	187
Tiled M-E CUDA	0.129	187

store the input and output tensors. Finally, the tiled memory efficient CUDA implementation further improves runtime. For all experiments we utilize the tiled memory efficient CUDA implementation.

Results are reported in Table 1. We can see that even the naive CUDA implementation offers a substantial improvement in both runtime and memory usage. Beyond this, the memory efficient CUDA implementation achieves the goal of peak memory usage no greater than that required to

## 4 An MVC/MVVS Architecture for Deformable Diffusion MRI Registration



**Fig. 1.** Deformable registration network architecture.

We now present architectures for unsupervised registration of dMRI data represented using fODF images (fODFs at each voxel). We present architectures for deformable (non-rigid) registration tasks. Several layers must be introduced. The first is the previously presented MVC/MVVS layer, which extract features from the fODF images. The second is a differentiable estimator of the local Jacobian of the deformation field. The third is a differentiable point spread function fODF (defined subsequently)

reorientation layer. Finally, we utilize the spatial transformer [12] layer for resampling of the fODF field. By using differentiable versions of the Jacobian estimation, reorientation and resampling operations we can perform the transformation of the moving fODF image inside the network, and compute a loss directly between the input fixed image and output warped image. This allows us to train in a fully unsupervised manner. This strategy has been used in several other methods for neural network based registration (e.g. [2, 28]). We release all of the code required for inference and training using these models.

#### 4.1 Differentiable Jacobian Estimator

When registering fODF images, a vital step is reorientation. When a deformation field is applied to an fODF image, the underlying spatial grid will be warped. Since the fODF functions at each voxel represent directional information, they must be transformed in accordance with the deformation field. We implement the reorientation method utilized in [17], which uses the local Jacobian of the deformation field to reorient the fODF functions.

Our model will be trained in an unsupervised manner, with the deformation and reorientation of the moving fODF image occurring inside the network during the training stage. Thus, we must compute the local Jacobian of the deformation field during training. To this end, we implement an efficient second order central difference based approximator for computing the Jacobian of the deformation field at each voxel. The 3D deformation field is represented as a tensor of shape  $B \times D \times W \times H \times 3$ , where  $B$ ,  $D$ ,  $W$ , and  $H$  are the batch, depth, width and height respectively. The Jacobian estimator computes a second order central difference estimation of the partial derivatives along each direction of the deformation vectors to output a  $B \times D \times W \times H \times 3 \times 3$  tensor, i.e. a field of local Jacobian matrices.

#### 4.2 Differentiable PSF Reorientation

Given the local Jacobian of the deformation field at every voxel, the next step is to reorient the fODF. Recall that the fODF at each voxel is represented as a density function on the sphere  $f : \mathbb{S}^2 \rightarrow \mathbb{R}$ . But we sample the fODF density along  $M$  sample points on the sphere to represent it as an  $M$ -dimensional probability vector. Thus our reorientation method must operate on this representation. We opt to implement a differentiable version of the method presented in [17]. In short, this method approximates the fODF function as a weighted sum of spherical point spread functions (PSFs), reorients the PSFs, and then resamples the weighted sum of the reoriented PSFs to return to the original representation, an  $M$ -directional probability vector. This method was shown to give improved results for fODF reorientation relative to previous methods, and satisfies some useful properties (e.g. the fODF integral and the partial volume fractions are guaranteed to be preserved).

We implement an efficient, batch-mode, differentiable version of this operation in PyTorch which can run on GPUs. To the best of our knowledge, this is the first differentiable implementation of an fODF reorientation method.

#### 4.3 Registration Architecture

We now have all the building blocks for an fODF registration network. A schematic of the proposed architecture is presented in Fig. 1. The moving and fixed images are concatenated along a channel axis and passed through a feature extraction head which consists of an MVC or MVVS based UNet, a ManifoldFC block and a traditional CNN based UNet. The ManifoldFC operation, originally introduced in [6], allows us to map manifold-valued features to scalar-valued features.

The output of the UNet heads is a deformation field (deformation vectors are stored across the channel dimension as in e.g. [2]). This deformation field is used to resample the input moving image. Simultaneously, the deformation field is passed through a Jacobian estimator block. The Jacobian matrices are then used to reorient the resampled moving image.

The MVC/MVVS UNet block consist of 4 layers mapping across the following channel sizes:  $2 \rightarrow 8 \rightarrow 16 \rightarrow 32 \rightarrow 32$ , each layer with a kernel size of 3. The traditional U-Net block consists of 3 encoder and 3 decoder layers, with a maximum of 1024 channels and skip connections between encoder and decoder layers.

In the deformable registration network a loss function is applied to the output warped image, with the input fixed image as the target as in [2]. By performing the resampling and reorientation inside the network, we can train in a fully unsupervised manner, in contrast to previous approaches such as [27] which required ground truth deformation fields to train.

## 5 Experiments

We now present several experiments demonstrating the performance of our MVC and MVVS registration networks on deformable registration tasks.

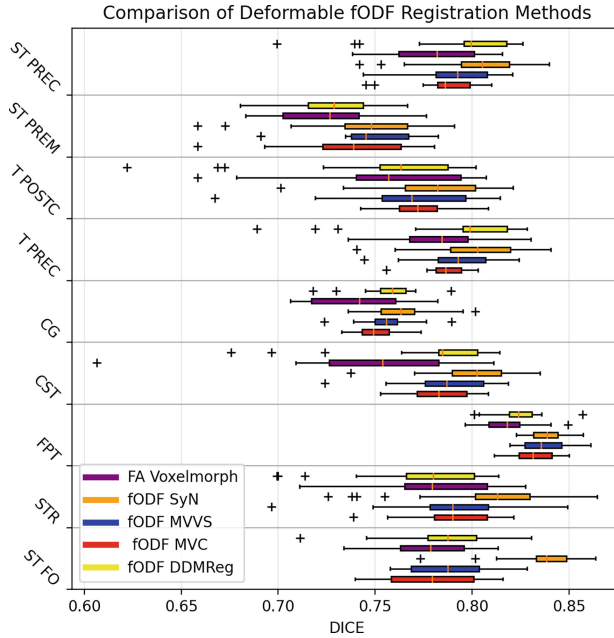
### 5.1 Dataset

We train and evaluate on dMRI data from the Human Connectome Project (HCP) Young Adult dataset. HCP consists of dMRI scans of the brain for 1200 subjects aged 22–35. For details about acquisition parameters, subject criteria, preprocessing etc. we refer the reader to the HCP study [21]. We randomly selected 400 subjects from the HCP dataset and run them through an fODF generation pipeline which consists of response function estimation using the technique in [9] to generate subject specific white matter, grey matter and CSF response functions. We then use multi-shell multi-tissue constrained spherical deconvolution [13] to reconstruct the white-matter fODF functions from the diffusion signal and estimated response function.

### 5.2 Evaluation Strategy

We evaluate registration accuracy by computing the DICE score between known fixed and warped structures. We limit our evaluation to white matter tracts, since this is a structure well captured by dMRI. Optimally, one would use expert labeled segmentation's of white matter tracts to perform the registration evaluation, but no such segmentation's exist for the HCP dataset. Instead, we opt to use a well-validated automatic segmentation algorithm to generate segmentation masks for white matter tracts in the moving and fixed image. Specifically, we utilize the TractSeg segmentation model [24]. A total of 72 white matter tracts were segmented. For subjects in the validation set, segmentation accuracy was visually reviewed and poor segmentation's were discarded. At evaluation time, a moving and fixed image pair are registered. The transformation is then used to transform each of the moving image white matter tract segmentation





**Fig. 2.** Comparison of deformable fODF registration methods on a subset of white matter tracts. The shortened tract names correspond to the following structures. ST PREC: Striato-precentral. ST PREM: Striato-premotor. T POSTC: Thalamo-postcentral. T PREC: Thalamo-precentral. CG: Cingulum left. CST: Corticospinal. FPT: Fronto-pontine. STR: Superior Thalamic Radiation. ST FO: Striato-fronto-orbital

masks. Finally, the DICE score is computed between the warped tract segmentation map and fixed tract segmentation mask.

For deformable registration, we focus on a template registration task. A randomly chosen subject from our 400 subject dataset is selected as a reference fixed image. All other samples from the dataset will be registered to this subject. Thus we have a dataset of 399 moving, fixed image pairs where the fixed image is the same for all samples. We split this dataset into 349 training samples, 25 validation samples and 25 test samples. Deformable registration algorithms generally require an accurate initialization to perform well. Thus, moving and fixed image pairs are first pre-aligned using an affine registration algorithm.

For this experiment, we take the additional step of downsampling the moving and fixed fODF images by a scale factor of  $1/2$  to obtain images of spatial dimensions  $64 \times 40 \times 64$ . In spite of the optimizations made in Sect. 2, fODF registration is a memory intensive task, thus downsampling was necessary to allow our registration networks to run on a single GPU. Future advances in hardware and algorithms will allow us to run our method on full resolution images.

We train for 1000 epochs using Adam optimizer with an initial learning rate of 0.0001 with the mean square error loss function. We select the model parameters at the epoch with the best validation set loss and use it for our evaluation.

Our set of comparison methods includes classical and deep learning based approaches. We start with a classical deformable registration approach. For this we use Symmetric Normalization (SyN) based deformable registration, which has been shown to be the state of the art across a variety of registration tasks [15]. We use the mrtrix3 implementation of SyN designed for fODF registration [18]. We configure the SyN registration algorithm to use a multi-resolution pyramid with 4 levels at scale factors of 1/8, 1/4, 1/2, and 1. We use a maximum number of iterations of 1000, 1000, 1000 and 100 at each level respectively, although an early stoppage criteria usually stops the registration before the maximum number of iterations. We also evaluate a Voxel-Morph style UNet FA registration model which can be obtained by removing the initial MVC/MVVS encoder head from the deformable registration architecture presented previously. We also test DDMReg, a deep learning based diffusion MRI registration method [28].

**Table 2.** Performance of tested deformable registration methods.

Method	DICE	Runtime (s)	% Voxels $\det(J) \leq 0$
FA Voxelmorph	0.7126	0.9	0.007%
fODF Classical	0.7601	839	0.012%
fODF MVVS	0.7493	2.6	0.006%
fODF MVC	0.7317	2.1	0.005%
fODF DDMReg	0.7417	12.2	0.003%

We measure the percentage of voxels in the deformation field with non-positive Jacobian determinants to determine regions where the deformation field is non-diffeomorphic. We found that for our methods, explicit regularization of generated deformation fields with a loss function

penalty term was not necessary. Indeed, our model achieves a very low percentage of voxels with non-positive Jacobians without an explicit regularization term.

### 5.3 Results

All evaluation results are computed across test sets unseen during training and not used for model checkpoint selection or hyperparameter optimization.

The deformable registration experiment results are presented in Table 2. Figure 2 shows the DICE performance of all compared methods on a subset of the 72 white matter tract structures. Again, DICE scores are averaged over all 72 white matter tract structures generated in our evaluation pipeline. Among the learning based (non-classical) techniques we again see the MVVS based model achieving the best DICE overlap score. The FA based VoxelMorph style model still lags behind all methods which use the full fODF. The MVVS based model again outperforms the MVC based model. The DDM-Reg based model is not trained end-to-end, instead opting for independently training several registration proposal networks and a registration fusion head. It also does not take the fODF as input directly, instead extracting hand-engineered features from the fODF image and inputting those features into the network (see Sect. 1.2 for details).

Optimally, the network should have the capacity to learn features useful for the registration task internally. We attribute the improved performance of our MVVS based model relative to the DDMReg model to these limitations of the DDMReg model, which are not present in our MVVS based fODF registration technique. Again, we approach the accuracy of the classical registration method at just 0.3% of the runtime. We see that the MVVS based model also outperforms DDMReg on registration time, in large part due to the optimizations made in our custom CUDA implementation of the MVVS layer Sect. 2. The runtime results in this table does not include preprocessing time required to generate the FA and TOM features required for DDMReg, which can take several additional minutes [28], thus our results underestimate the true runtime improvement of our MVVS based model over DDMReg. Finally, all methods achieve a very small percentage of deformation field voxels with non-positive Jacobian, indicating that generated deformation fields are close to being diffeomorphic.

## 6 Conclusions

In this paper we presented a novel geometric deep neural network for registration of fODF images. We presented a registration model that respects the underlying geometry of fODF (manifold-valued) images. We also presented an efficient CUDA implementation of the vital manifold-valued image processing layer (MVC/MVVS) and introduced a novel Jacobian estimation and reorientation layer. Overall, our method is the first end-to-end trained model for fODF (dMRI) image registration. Finally, we presented several experiments demonstrating that our MVVS (MVC) for deformable registration achieve accuracy in par with classical methods but at a fraction of the processing time.

**Acknowledgements.** This research was in part supported by the NSF grant IIS-1724174, the NIH NINDS and NIA via RF1NS121099 to Vemuri and the MOST grant 110-2118-M-002-005-MY3 to Yang.

## References

1. Avants, B.B., Tustison, N., Song, G.: Advanced normalization tools (ANTs). *Insight j* 2(365), 1–35 (2009)
2. Balakrishnan, G., Zhao, A., Sabuncu, M.R., et al.: An unsupervised learning model for deformable medical image registration. In: *Proceedings of the IEEE CVPR*, pp. 9252–9260 (2018)
3. Barmpoutis, A., Vemuri, B.C.: Groupwise registration and atlas construction of 4th-order tensor fields using the R+ Riemannian metric. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) *MICCAI 2009*. LNCS, vol. 5761, pp. 640–647. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-04268-3\\_79](https://doi.org/10.1007/978-3-642-04268-3_79)
4. Bouza, J.J.: Manifold valued Volterra CNNs with apps. *Med. Img.* Ph.D. thesis, University of Florida (2022)
5. Bouza, J.J., Yang, C.-H., Vaillancourt, D., Vemuri, B.C.: A higher order manifold-valued convolutional neural network with applications to diffusion MRI processing. In: Feragen, A., Sommer, S., Schnabel, J., Nielsen, M. (eds.) *IPMI 2021*. LNCS, vol. 12729, pp. 304–317. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-78191-0\\_24](https://doi.org/10.1007/978-3-030-78191-0_24)

6. Chakraborty, R., Bouza, J., Manton, J., Vemuri, B.C.: ManifoldNet: a deep neural network for manifold-valued data with applications. *IEEE TPAMI* **44**(2), 799–810 (2022)
7. Cheng, G., Vemuri, B.C., Carney, P.R., Mareci, T.H.: Non-rigid registration of high angular resolution diffusion images represented by Gaussian mixture fields. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) *MICCAI 2009*. LNCS, vol. 5761, pp. 190–197. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-04268-3\\_24](https://doi.org/10.1007/978-3-642-04268-3_24)
8. Dalca, A.V., Balakrishnan, G., Guttag, J., Sabuncu, M.R.: Unsupervised learning of probabilistic diffeomorphic registration for images and surfaces. *Med. Img. Anal.* **57**, 226–236 (2019)
9. Dhollander, T., Raffelt, D., Connelly, A.: Unsupervised 3-tissue response function estimation from single-shell or multi-shell diffusion MR data without a co-registered T1 image. In: *ISMRM Workshop on Breaking the Barriers of Diffusion MRI*, vol. 5. ISMRM (2016)
10. Henrik, J.G.: Density-based similarity in the registration of diffusion-weighted images. Ph.D. thesis, University of Copenhagen (2018)
11. Hoffmann, M., Billot, B., Greve, D.N., Iglesias, J.E., Fischl, B., Dalca, A.V.: SynthMorph: learning contrast-invariant registration without acquired images. *arXiv preprint arXiv:2004.10282* (2020)
12. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: *Advances in NeurIPS*, pp. 2017–2025 (2015)
13. 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)
14. Jones, D., Lewis, D., Alexander, D., et al.: Spatial normalization and averaging of diffusion tensor MRI data sets. *Neuroimage* **17**(2), 592–617 (2002)
15. Klein, A., Andersson, J., Ardekani, B.A., et al.: Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. *Neuroimage* **46**(3), 786–802 (2009)
16. Li, H., Fan, Y.: Non-rigid image registration using fully convolutional networks with deep self-supervision. *arXiv preprint arXiv:1709.00799* (2017)
17. Raffelt, D., Tournier, J.D., Crozier, S., Connelly, A., Salvado, O.: Reorientation of fiber orientation distributions using apodized point spread functions. *MRM* **67**(3), 844–855 (2012)
18. Raffelt, D., Tournier, J.D., Fripp, J., et al.: Symmetric diffeomorphic registration of fibre orientation distributions. *Neuroimage* **56**(3), 1171–1180 (2011)
19. 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)
20. 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)
21. Van Essen, D.C., Ugurbil, K., Auerbach, E., et al.: The human connectome project: a data acquisition perspective. *Neuroimage* **62**(4), 2222–2231 (2012)
22. Vemuri, B.C., Sun, J., Banerjee, M., et al.: A geometric framework for ensemble average propagator reconstruction from diffusion MRI. *Med. Img. Anal.* **57**, 89–105 (2019)
23. de Vos, B.D., Berendsen, F.F., Viergever, M.A., Staring, M., Išgum, I.: End-to-end unsupervised deformable image registration with a convolutional neural network. In: Cardoso, M.J., et al. (eds.) *DLMIA/ML-CDS 2017*. LNCS, vol. 10553, pp. 204–212. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-67558-9\\_24](https://doi.org/10.1007/978-3-319-67558-9_24)
24. Wasserthal, J., Neher, P., Maier-Hein, K.H.: TractSeg-fast and accurate white matter tract segmentation. *Neuroimage* **183**, 239–253 (2018)
25. van Werkhoven, B., Maassen, J., Bal, H.E., Seinstra, F.J.: Optimizing convolution operations on GPUs using adaptive tiling. *Future Gener. Comput. Syst.* **30**(C), 14–26 (2014)

26. Yang, J., Shen, D., Davatzikos, C., Verma, R.: Diffusion tensor image registration using tensor geometry and orientation features. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (eds.) MICCAI 2008. LNCS, vol. 5242, pp. 905–913. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-85990-1\\_109](https://doi.org/10.1007/978-3-540-85990-1_109)
27. Yang, X., Kwitt, R., Styner, M., Niethammer, M.: Quicksilver: fast predictive image registration—a deep learning approach. *Neuroimage* **158**, 378–396 (2017)
28. Zhang, F., Wells, W.M., O’Donnell, L.J.: Deep diffusion MRI registration (DDMReg): a deep learning method for diffusion MRI registration. *IEEE Trans. Med. Img.* (2021)
29. Zhang, P., Niethammer, M., Shen, D., Yap, P.-T.: Large deformation diffeomorphic registration of diffusion-weighted images with explicit orientation optimization. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013. LNCS, vol. 8150, pp. 27–34. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-40763-5\\_4](https://doi.org/10.1007/978-3-642-40763-5_4)