



Explicit B-spline regularization in diffeomorphic image registration

Nicholas J. Tustison^{1*} and Brian B. Avants²

¹ Department of Radiology and Medical Imaging, University of Virginia, Charlottesville, VA, USA

² Penn Image Computing and Science Laboratory, Department of Radiology, University of Pennsylvania, Philadelphia, PA, USA

Edited by:

Hans J. Johnson, The University of Iowa, USA

Reviewed by:

Ipek Oguz, UNC Chapel Hill, USA
Andrey Fedorov, Brigham and Women's Hospital, USA

*Correspondence:

Nicholas J. Tustison, Department of Radiology and Medical Imaging, University of Virginia, 480 Ray C Hunt Drive, Suite 124, Charlottesville, VA 22903, USA
e-mail: ntustison@virginia.edu

Diffeomorphic mappings are central to image registration due largely to their topological properties and success in providing biologically plausible solutions to deformation and morphological estimation problems. Popular diffeomorphic image registration algorithms include those characterized by time-varying and constant velocity fields, and symmetrical considerations. Prior information in the form of regularization is used to enforce transform plausibility taking the form of physics-based constraints or through some approximation thereof, e.g., Gaussian smoothing of the vector fields [à la Thirion's Demons (Thirion, 1998)]. In the context of the original Demons' framework, the so-called *directly manipulated free-form deformation (DMFFD)* (Tustison et al., 2009) can be viewed as a smoothing alternative in which explicit regularization is achieved through fast B-spline approximation. This characterization can be used to provide B-spline "flavored" diffeomorphic image registration solutions with several advantages. Implementation is open source and available through the Insight Toolkit and our Advanced Normalization Tools (ANTs) repository. A thorough comparative evaluation with the well-known SyN algorithm (Avants et al., 2008), implemented within the same framework, and its B-spline analog is performed using open labeled brain data and open source evaluation tools.

Keywords: Advanced normalization tools, diffeomorphisms, directly manipulated free-form deformation, Insight Toolkit, spatial normalization

1. INTRODUCTION

Establishment of anatomical and functional correspondence is a crucial step toward gaining insight into biological processes. Neuroscience research efforts, such as characterizing brain morphology, require accurate and robust methods for producing such mappings. The extensive literature detailing methodology is evidence of the rich history of algorithmic development which continues contemporaneously. We highlight several key historical contributions which are particularly relevant to the work presented.

Free-form deformation (FFD) image registration, characterized by regularization based on the B-spline basis functions, has several advantages including algorithmic simplicity, good performance, and guaranteed parametric continuity. Current research was preceded by related work for geometric modeling (Sederberg and Parry, 1986) and originated with such important contributions as Szeliski and Coughlan (1997); Thévenaz et al. (1998), and Rueckert et al. (1999). Continued development within this early spline-based paradigm produced additional innovations such as integrated similarity metrics (e.g., Mattes et al., 2003), additional transformation constraints (e.g., Rohlfing et al., 2003), and notable open source implementations (e.g., Ibanez et al., 2005; Klein et al., 2010b; Modat et al., 2010; Shackleford et al., 2010).

Parallel to this branch of algorithmic progress are the informally denoted "dense transforms" perhaps best exemplified by Thirion's seminal contribution (Thirion, 1998). Relationships with earlier elastic (Bajcsy and Kovacic, 1989; Gee et al., 1993) and fluid (Christensen et al., 1996) registration methods are

detailed in the works of Bro-Nielsen and Gramkow (1996) and Pennec et al. (1999) who observe that smoothing via Gaussian convolution, a defining characteristic of Demons, of the update or total displacement field is a greedy approximation for solving the partial differential equations governing the physics of an elastic or fluid deformation, respectively. However, the use of such approximations entails that physical properties, such as topological regularity, are no longer guaranteed.

It is interesting to note that within this context, traditional FFD algorithms can be viewed as a type of fluid-like Demons approach where, rather than projecting the update field to the space of regularized fields using Gaussian convolution, gradient fields are projected to a smooth space characterized by the B-spline basis functions. This analogy was hinted at in our earlier work (Tustison et al., 2009) where we showed that fitting the update field to a B-spline object using a fast approximation routine (Tustison and Gee, 2006) is equivalent to a preconditioning of the standard gradient used in gradient descent-based FFD optimization. This preconditioning is used to mitigate the hemstitching effect induced by the ill-conditioned nature of the traditional gradient-based FFD formulation¹. We denoted this new FFD variant as *directly manipulated free-form deformation*

¹This phenomenon is explained in greater detail in Tustison et al. (2009) but, briefly, the distribution of the uniform B-spline basis functions spanning the image domain produces long ravines in the image registration energy topography. These topographical features cause an inefficient back-and-forth traversal in classic steepest descent optimization, also known as "hemstitching."

(DMFFD) and, as part of the ITKv4 refactoring efforts, has been implemented for use with the new registration framework² which permits both B-spline smoothing on the update (“viscous”) and total (“elastic”) displacement fields at each iteration (cf analogous Gaussian, i.e., Demons, implementation³).

Continuing from the work of Christensen et al. (1996) and subsequent exploration into the mathematical formalisms of diffeomorphisms (e.g., Dupuis and Grenander, 1998), the well-known Large Deformation Diffeomorphic Metric Mapping (LDDMM) algorithm was proposed in Beg et al. (2005). In contrast to the mapping produced by Christensen et al. (1996), LDDMM yields the geodesic solution in the space of diffeomorphisms between two images. Since its introduction, LDDMM has inspired much innovation in the image registration literature. Applying the log-Euclidean framework of Arsigny et al. (2006), DARTEL (Diffeomorphic Anatomic Registration using Exponential Lie algebra) uses a constant velocity field parameterization to provide a fast, diffeomorphic alternative (Ashburner, 2007). Additionally, symmetrical considerations in the velocity field parameterization are discussed in Avants et al. (2008) in the context of a cross correlation similarity metric. By explicitly symmetrizing the LDDMM formulation, this Symmetric Normalization (SyN) approach minimizes the bias of the resulting transformation when selecting the “fixed” and “moving” images. A greedy version of this algorithm has proven successful in neuroimaging (Klein et al., 2009) and pulmonary (Murphy et al., 2011) applications as well as in multi-atlas label fusion (Wang et al., 2013).

Although many extensions of LDDMM rely on some form of Gaussian convolution for regularization (e.g., Risser et al., 2011), there has been significant interest in constraining FFD approaches to the space of diffeomorphisms. An early attempt reported in Rueckert et al. (2006) enforced diffeomorphic transforms by concatenating multiple FFD transforms, each of which is constrained to describe a one-to-one mapping. Modat et al. (2011) incorporated the log-Euclidean framework for enforcing diffeomorphic transformations and ensuring invertibility. Similarly, the work of De Craene et al. (2011) provided a full LDDMM-style algorithm based on B-splines called *temporal FFD* in which the time-varying velocity field is modeled using a 4-D B-spline object (3-D + time). Numerical Eulerian integration of the mapping propagated within the velocity field yields the transform between parameterized time points.

As alluded to earlier, B-spline approximation can also be used for regularizing time-varying vector fields in an analogous fashion as Gaussian convolution. In this vein, and similar to De Craene et al. (2011), we reported in Tustison and Avants (2012a,b) the use of an n -D + time B-spline object to represent the characteristic velocity fields. However, we use the DMFFD formulation to improve the solution convergence. This also facilitates modeling temporal periodicity and the enforcement of stationary boundaries. Both this work and our earlier work (Tustison et al., 2009)

demonstrate that the DMFFD framework is potentially applicable to the entire gamut of diffeomorphic registration algorithms and provides alternative smoothing possibilities with different continuity properties (e.g., C^2 vs. C^3).

The two regularization approaches (i.e., Gaussian convolution vs. DMFFD), however, produce characteristically different solutions. In addition to smoothing kernel differences, Gaussian convolution tends to “flatten” the signal in contrast to an approximation or fitting of the signal provided by the DMFFD approach. Also, whereas Gaussian convolution operates entirely within discretized space, the B-spline approximation routine constructs a continuous object prior to any voxelwise reconstruction of the sampled fields. Interestingly, a similar comparison was made with respect to Gaussian derivative estimation (Bouma et al., 2007). Although typically estimated using truncated, discrete Gaussian convolution, an alternative based on B-spline approximation demonstrated superior performance with similar computational cost.

Three popular diffeomorphic algorithms and their DMFFD analogs (LDDMM, DARTEL, and SyN) were implemented by the authors as part of the recent refactoring of the open source Insight Toolkit (ITK) although related work had been previously implemented within the popular Advanced Normalization Tools (ANTs)⁴. Given the popularity and excellent performance of the greedy variant of SyN, our evaluation focus in this work is its B-spline analog which we denote as “B-spline SyN” or “DMFFD SyN.” Evaluations of the respective algorithmic instantiations are performed using the `antsRegistration` program found in the ANTs repository (also originally developed by the authors). This permits a direct algorithmic comparison as potential sources for implementation bias have been reduced (Tustison et al., 2013). Additionally, in the spirit of open science, all text, figures, and scripts to reproduce the results contained in this work are publicly available online⁵.

2. MATERIAL AND METHODS

2.1. THEORETICAL OVERVIEW

Given the spatial domain Ω of d -dimensionality defined over image I , a diffeomorphic mapping, ϕ , parameterized over $t \in [0, 1]$ transforms the image I to the target image J using $I \circ \phi(\mathbf{x}, t)$ where the geodesic path $\phi(\mathbf{x}, t)$ is described by (Beg et al., 2005)

$$\inf_{\phi} \left(\int_0^1 \|\nu(t)\|_L^2 dt + \int_{\Omega} |I \circ \phi^{-1}(\mathbf{x}, 1) - J|^2 d\Omega \right). \quad (1)$$

ϕ is generated as the solution of the ordinary differential equation

$$\frac{d\phi(\mathbf{x}, t)}{dt} = \nu(\phi(\mathbf{x}, t), t), \quad \phi(\mathbf{x}, 0) = \mathbf{Id} \quad (2)$$

where ν is a time-dependent smooth field (as dictated by the functional norm L), $\nu: \Omega \times t \rightarrow \mathbb{R}^d$. Diffeomorphic mappings between parameterized time points $\{t_a, t_b\} \in [0, 1]$ are obtained

²http://www.itk.org/Doxygen/html/classitk_1_1BSplineSmoothingOnUpdateDisplacementFieldTransform.html.

³http://www.itk.org/Doxygen/html/classitk_1_1GaussianSmoothingOnUpdateDisplacementFieldTransform.html.

⁴<http://stnava.github.io/ANTs>.

⁵<https://github.com/ntustison/BSplineMorphisms>.

from Equation (2) through integration of the transport equation, viz.

$$\phi(x, t_b) = \phi(x, t_a) + \int_{t_a}^{t_b} v(\phi(x), t) dt. \quad (3)$$

However, as pointed out in Avants et al. (2008), implementations of this standard LDDMM formulation are negatively affected by the lack of optimization symmetry where arbitrary assignment of fixed and moving images could lead to different solutions despite the fact that the theoretical geodesic solution describes the same parameterized path forwards and backwards. This observation led to the symmetric formulation of Equation (1) found in Avants et al. (2008):

$$\inf_{\phi_1} \inf_{\phi_2} \left(\int_0^{0.5} (\|v_1(t)\|_L^2 + \|v_2(t)\|_L^2) dt + \int_{\Omega} |I \circ \phi_1^{-1}(x, 0.5) - J \circ \phi_2^{-1}(x, 0.5)|^2 d\Omega \right) \quad (4)$$

where

$$\frac{d\phi_i(x, t)}{dt} = v_i(\phi_i(x, t), t), \quad \phi_i(x, 0) = Id, \quad i \in \{1, 2\}. \quad (5)$$

With extension to arbitrary similarity metric choice, the second term is replaced with

$$\int_{\Omega} \Pi_{\sim} \left(I \circ \phi_1^{-1}(x, 0.5), J \circ \phi_2^{-1}(x, 0.5) \right) d\Omega \quad (6)$$

with a popular choice for Π_{\sim} being a local neighborhood cross correlation (Avants et al., 2008, 2011). Note that t is parameterized in opposite directions between ϕ_1 and ϕ_2 . A diagrammatic illustration of the explicit symmetry associated with SyN is shown in Figure 1.

2.1.1. St. nava's theory of greed and original SyN

Although presenting a rigorous framework for image registration solutions with desirable properties, the complexity of these diffeomorphic methodologies requires substantial computational resources. For typical 3-D neuroimaging applications, the corresponding solutions require numerical integration over and storage of 4-D velocity fields at each iteration which is limiting for many common computational platforms.

Therefore, in addition to the full-scale SyN offering described in Avants et al. (2008), the authors therein provided a “greedy” alternative which has demonstrated superior performance in different applications (Klein et al., 2009; Avants et al., 2011; Murphy et al., 2011)⁶ while simultaneously being capable of running with limited computational resources. This is due to the restriction of the discrete, time-parameterized velocity field samples to their respective endpoints, i.e., the $v_i(x, t)$ are sampled at $t \in$

$\{0, 0.5, 1\}$ implying simultaneous storage of only four transform vector fields $\phi_1(x)$, $\phi_1^{-1}(x)$, $\phi_2(x)$, and $\phi_2^{-1}(x)$ (cf Figure 1). Furthermore, the forward and inverse mappings are guaranteed to be consistent within the discrete domain i.e., $\|\phi_i^{-1}(\phi_i) - Id\|^2 < \epsilon$. Since the greedy SyN framework is the focus of the evaluation, the algorithmic steps are briefly sketched in Algorithm 1.

Algorithm 1 | Greedy SyN algorithm

```

 $\phi_i \leftarrow Id, \phi_i^{-1} \leftarrow Id \quad \triangleright i \in \{1, 2\}$ 
for all image resolution levels do
   $n \leftarrow 1$ 
  while not converged do
     $v_1^n \leftarrow \nabla \Pi_{\sim} \left( I \circ \phi_1^{n-1}, J \circ \phi_2^{n-1} \right)$ 
     $v_2^n \leftarrow \nabla \Pi_{\sim} \left( J \circ \phi_2^{n-1}, I \circ \phi_1^{n-1} \right)$ 
     $v_i^n \leftarrow S_v(v_i^n) \quad \triangleright S_v$  is a smoothing operation on the
    update transform field
     $\phi_i^n \leftarrow S_{\phi}(v_i^n \circ \phi_i^{n-1}) \triangleright S_{\phi}$  is a smoothing operation on
    the total transform field
     $(\phi_i^n)^{-1} \leftarrow Inv \left( \phi_i^n, (\phi_i^{n-1})^{-1} \right) \quad \triangleright$  Inverse field
    estimation described in Avants et al. (2008)
     $n \leftarrow n + 1$ 
  end while
  upsample current  $\phi_i$  and  $\phi_i^{-1}$  to next resolution level  $\triangleright$ 
   $i \in \{1, 2\}$ 
end for
return  $\phi \leftarrow \phi_1 \circ \phi_2^{-1}, \phi^{-1} \leftarrow \phi_2 \circ \phi_1^{-1}$ 

```

2.1.2. Directly manipulated free-form deformation diffeomorphic analogs

Although several velocity field regularization operators have been proposed, many algorithmic instantiations default to Gaussian smoothing due to its simplicity both in implementation and complexity terms. A viable and practical alternative is the DMFFD approach based on B-splines for explicit regularization of vector fields.

Given the similarity metric Π_{\sim} , the d -dimensional update field (i.e., preconditioned gradient field), $\delta v_{i_1, \dots, i_d}$, is given by

$$\delta v_{i_1, \dots, i_d} = \frac{\sum_{c=1}^{N_{\Omega}} \left(\frac{\partial \Pi_{\sim}}{\partial \mathbf{x}} \right)_c \prod_{j=1}^d B_{i_j}(x_j^c) \cdot \frac{\prod_{j=1}^d B_{i_j}^2(x_j^c)}{\sum_{k_1=1}^{r+1} \dots \sum_{k_d=1}^{r+1} \prod_{j=1}^d B_{i_j}^2(x_j^c)}}{\sum_{c=1}^{N_{\Omega}} \prod_{j=1}^d B_{i_j}^2(x_j^c)} \quad (7)$$

where the set of $B(\cdot)$ are the univariate B-spline basis functions for separately modulating regularity in the solution for each parametric dimension, N_{Ω} is the number of voxels in the reference image domain, r is the spline order in all dimensions,⁷ and $\frac{\partial \Pi_{\sim}}{\partial \mathbf{x}}$ is the spatial similarity metric gradient at voxel c .⁸

⁷In terms of implementation, spline orders can be specified separately for each dimension but, for simplicity, we only specify a single spline order.

⁸For comparison, FFD image registration uses the calculated gradient which, for a single control point is evaluated using (in discretized form assuming no

⁶SyN was also the standard registration for the MICCAI 2013 Workshop on Segmentation: Algorithms Theory, and Applications (SATA) and corresponding challenge (<https://masi.vuse.vanderbilt.edu/workshop2013>).

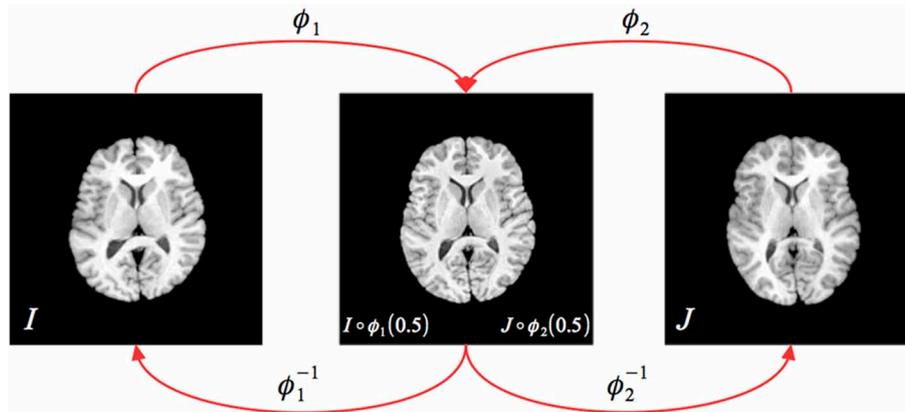


FIGURE 1 | Illustration of the greedy SyN formulation. Given images I_A and I_B , the symmetric set-up requires finding the two transform pairs (ϕ_1, ϕ_1^{-1}) (ϕ_2, ϕ_2^{-1}) which map to/from the respective images to the midway point. During optimization, the update field at each

iteration is determined from the metric field gradient taken at the midway point, i.e., $\nabla \Pi_{\sim}(I \circ \phi_1(0.5), J \circ \phi_2(0.5))$. The full forward and inverse transforms are found through composition, i.e., $\phi = \phi_1 \circ \phi_2^{-1}$ and $\phi^{-1} = \phi_2 \circ \phi_1^{-1}$.

Similarly, in the case of d -dimensional time-parameterized diffeomorphic image registration, the time-dependent velocity field can be represented as a $(d + 1)$ -dimensional B-spline object

$$v(x, t) = \sum_{i_1=1}^{X_1} \dots \sum_{i_d=1}^{X_d} \sum_{i_t=1}^T v_{i_1, \dots, i_d, i_t} B_{i_t}(t) \prod_{j=1}^d B_{i_j}(x_j) \quad (9)$$

other explicit regularization):

$$\delta v_{i_1, \dots, i_d} = \sum_{c=1}^{N_{\Omega}} \left(\frac{\partial \Pi_{\sim}}{\partial \mathbf{x}} \right)_c \prod_{j=1}^d B_{i_j}(x_j^c). \quad (8)$$

Intuitively, this can be understood as a distribution of the spatial similarity metric gradient to the set of control points according to the basis function weighting which, due to the local property of B-splines, is only non-zero in a localized region surrounding each sample point. The problematic issue with standard FFD image registration is that the parametric domain of the peripheral B-spline basis functions extends beyond the image boundaries resulting in a relatively lower weighting contribution effect. As a corrective, Equation (7) normalizes the control point gradient contribution by the actual B-spline weighting overlap with the image domain.

Note that this modification is first discussed and derived in Hsu et al. (1992) and Lee et al. (1997) in the context of fitting scattered data for a computer graphics audience which was an extension and improvement over Sederberg's original FFD manipulation computer graphics technique (Sederberg and Parry, 1986). Whereas the latter (i.e., FFD) technique performs geometric deformation of the graphics object embedded in a B-spline object via manipulation of the control points, the former (i.e., DMFFD) technique permits direct manipulation of the actual object with the control point values being updated indirectly. The corresponding fast algorithm for scattered data approximation using DMFFD was proposed in Lee et al. (1997).

Based on the needs of one of our colleagues, the first author (N.T.) implemented and generalized the work of Lee et al. (1997) which was reported in Tustison and Gee (2006) and was N.T.'s very first ITK contribution. A couple years later the second author (B.A.) requested an FFD implementation for ANTs which N.T. naively assumed to be equivalent to applying Tustison and Gee (2006) to smoothing the metric gradient (a la Demons). The subsequent realization that this assumption was incorrect and figuring out why the algorithm still worked despite it not being a "correct" FFD implementation resulted in Tustison et al. (2009).

where v_{i_1, \dots, i_d, i_t} is a $(d + 1)$ -dimensional control point lattice characterizing the velocity field. The preconditioned gradient analog of Equation (7) for updating the time-varying velocity field control point lattice is

$$\delta v_{i_1, \dots, i_d, i_t} = \left(\sum_{c=1}^{N_{\Omega} \times N_t} \left(\frac{\partial \Pi_{\sim}}{\partial \mathbf{x}} \right)_c B_{i_t}(t^c) \prod_{j=1}^d B_{i_j}(x_j^c) \right) \cdot \frac{B_{i_t}^2(t^c) \prod_{j=1}^d B_{i_j}^2(x_j^c)}{\sum_{k_1=1}^{r+1} \dots \sum_{k_d=1}^{r+1} \sum_{k_t=1}^{r+1} B_{k_t}^2(t^c) \prod_{j=1}^d B_{k_j}^2(x_j^c)} \cdot \left(\sum_{c=1}^{N_{\Omega} \times N_t} B_{i_t}^2(t^c) \prod_{j=1}^d B_{i_j}^2(x_j^c) \right)^{-1} \quad (10)$$

which takes into account the temporal locations of the dense gradient field sampled in $t \in [0, 1]$. N_t and N_{Ω} are the number of time point samples and the number of voxels in the reference image domain, respectively⁹.

For regularization of constant velocity fields, e.g., SyN or DARTEL, updating the field is performed using Equation (7). In the case B-spline SyN, this applies to the smoothing operators S_v and S_{ϕ} in **Algorithm 1** although best performance (at least for the data described in this work) typically employs no smoothing on the total transform field, i.e., S_{ϕ} is such that $S_{\phi}(\phi_i^n) = \phi_i^n$.

2.2. IMPLEMENTATION

Three diffeomorphic algorithms described previously utilizing Gaussian convolution smoothing and their DMFFD counterparts are available through the following classes in the ITK repository:

- LDDMM:
 - `itk::TimeVaryingVelocityFieldTransform`

⁹Additionally, in Tustison and Gee (2006) it was shown that one could associate each a sample, $\left(\frac{\partial \Pi_{\sim}}{\partial \mathbf{x}} \right)_c$ with a confidence weighting. Thus, in order to enforce stationary boundaries for all described DMFFD-based diffeomorphic algorithms, we assign to the image boundary metric gradients the null vector with a corresponding large confidence value.

- `itk::TimeVaryingVelocityFieldTransformParametersAdaptor`
 - `itk::TimeVaryingVelocityFieldImageRegistrationMethodv4`
 - `itk::GaussianSmoothingOnUpdateTimeVaryingVelocityFieldTransform`
 - `itk::TimeVaryingBSplineVelocityFieldTransform`
 - `itk::TimeVaryingBSplineVelocityFieldImageRegistrationMethodv4`
 - `itk::TimeVaryingBSplineVelocityFieldTransformParametersAdaptor`
- **DARTEL:**
 - `itk::ConstantVelocityFieldTransform`
 - `itk::ConstantVelocityFieldTransformParametersAdaptor`
 - `itk::GaussianExponentialDiffeomorphicTransform`
 - `itk::GaussianExponentialDiffeomorphicTransformParametersAdaptor`
 - `itk::BSplineExponentialDiffeomorphicTransform`
 - `itk::BSplineExponentialDiffeomorphicTransformParametersAdaptor`
 - **SyN:**
 - `itk::SyNImageRegistrationMethod`
 - `itk::BSplineSyNImageRegistrationMethod`

These and other classes (e.g., similarity metrics, optimization methods, and utility classes) were developed as part of the ITKv4 registration framework refactoring. Much of the original ITK image registration infrastructure was left intact including the so-called “sparse” transforms such as various rigid (versor, Euclidean) and other linear transforms. The transform classes contributed by our group were, including those listed above, were meant to augment what already existed. All the transforms listed above are derived from the `itk::DisplacementFieldTransform` class which permits specification of a transform described by a sampled displacement field. The derived classes are then modified according to the different transform constraints. Other types of classes are used to coordinate the image registration process. The `itk::ImageRegistrationMethodv4` is the base interface for performing all image registration steps. Smoothing and resampling for multi-resolution image registration is performed in this class as is the calling of the selected optimizer. Output consists of a single optimized transform. Multiple instantiations of this class in series, in conjunction with the `itk::CompositeTransform` class, are used to optimize a composition of transforms. Some of the diffeomorphic approaches do not easily fit into this generalized optimization framework necessitating specialized method classes such as those

listed above. The adaptors¹⁰ are used to modify the parameters between resolution levels during the course of transform optimization within the methods classes. For example, the resolution of the displacement field transforms follows that of the image resolution and the updating is handled by the corresponding transform adaptors. Further details can be found in the documentation provided within the classes themselves¹¹.

The class `itk::BSplineScatteredDataPointSetToImageFilter` underlies all DMFFD regularization which is an implementation of the methods described in Tustison and Gee (2006). Although applicable to various scenarios (e.g., curve and surface estimation), it has been optimized for imaging applications and multi-threaded for fast processing on suitable machines. Additionally, numerical integration for solving Equations (2) and (3) utilizes Runge-Kutta which provides a more stable alternative than other methods (Press et al., 2007). Implementation is provided in the class `itk::TimeVaryingVelocityFieldIntegrationImageFilter`.

A complete packaging of these classes has been made available as part of our ANTs toolkit¹². The `antsRegistration` program¹³ takes advantage of the enhanced ITKv4 registration framework and was developed by the authors to provide a robust and versatile solution for a wide variety of image registration applications. The basic conceptualization for use is that one can set-up any number of registration “stages” with each stage being characterized by a specified transform. For example, a representative command call is as follows:

```
# Register the $fixed and $moving images
# with initial alignment of the centers
# of intensity followed by the following
# three stages:
# rigid -> affine -> B-spline SyN

antsRegistration
  --dimensionality 3 \
  --output ${prefix} \
  --use-histogram-matching 1 \
  --initial-moving-transform [${fixed},
    ${moving},1] \
  --transform Rigid[0.1] \
  --metric MI[${fixed},${moving},1,32,
    Regular,0.25] \
  --convergence 1000x500x250x100 \
  --smoothing-sigmas 3x2x1x0 \
  --shrink-factors 8x4x2x1 \
  --transform Affine[0.1] \
```

¹⁰We gratefully acknowledge Marius Staring’s contribution in providing the broad design of the adaptors.

¹¹<http://www.itk.org/Doxygen/html/classes.html>.

¹²<http://stnava.github.io/ANTs/>.

¹³The long and short command line help menus can be invoked by the commands ‘`antsRegistration -help`’ and ‘`antsRegistration -h`’, respectively.

```

--metric MI[${fixed},${moving},1,32,
  Regular,0.25] \
--convergence 1000x500x250x100 \
--smoothing-sigmas 3x2x1x0 \
--shrink-factors 8x4x2x1 \
--transform BSplineSyN[0.1,26,0,3] \
--metric CC[${fixed},${moving},1,4] \
--convergence 100x70x50x20 \
--smoothing-sigmas 3x2x1x0 \
--shrink-factors 6x4x2x1

# Apply the resulting transforms (generic
# affine + B-spline SyN) to the
# moving labels.

antsApplyTransforms
  --dimensionality 3 \
  --input ${moving_labels} \
  --reference-image ${fixed} \
  --output ${moving_warped_labels} \
  --n NearestNeighbor \
  --transform ${prefix}1Warp.nii.gz \
  --transform ${prefix}0GenericAffine.mat \
  --default-value 0

```

Listing 1 | Representative script containing antsRegistration and antsApplyTransforms command calls used for evaluation.

In this example, we first calculate an initial translation transform by aligning the centers of (intensity) mass (although alignment based on other features is possible)¹⁴. The resulting transform is then used as input for determining an optimal rigid transform. Serial propagation of the resulting composite transform continues until all optimal transforms have been determined. Optimization for each stage is determined by the specified general parameters including: smoothing and downsampling of fixed and moving images, convergence criteria (including number of iterations per resolution level) and metric (or metrics). Any pair of images can be specified per metric per stage¹⁵.

Although the resulting transforms for each stage can be written to disk as output, the default output consists of a condensed set of transforms where compatible transforms have been composed to a single transform. For example, in the above command call, the initial translation, rigid, and affine transforms are combined into a single generic affine transform file with the results of the deformable transform consisting of discrete vector

¹⁴This step corresponds to the `-initial-moving-transform` option. Generally, the user can specify an ITK transform for initialization or can perform an initial translation based on either the geometric center of the images, the center of the image intensities, or the origin of the images. See the help menu `'antsRegistration -help'` for more details.

¹⁵To help the reader who wishes to explore `antsRegistration` with specific use of the parameters used for this study and specified above, we have created a script with a simplified interface and placed it in the `Scripts/` subdirectory of the ANTs repository. This script, called `antsRegistrationSyN.sh` takes a fixed and moving image and performs a comprehensive (i.e., rigid → affine → deformable) registration using SyN (`-t s`) or B-spline SyN (`-t b`).

fields. The output transform files can then be applied using the `antsApplyTransforms` program which permits composition of any number of transform files with different interpolation schemes. For both programs interpolation is never performed more than once.

2.3. EVALUATION DATA

In the well-known Klein comparative study (Klein et al., 2009), 14 image registration algorithms were evaluated based on performance on publicly available labeled brain data. For our evaluation, we used these same data. Specifically, we used the data sets denoted as:

- CUMC12
- IBSR18
- LPBA40
- MGH10

which are available for download from Arno Klein's website¹⁶.

The number of subjects per cohort is provided in the denotation. **Table 1** summarizes core information about the data sets used. Further details of these first four labeled brain data (e.g., labeling protocol, data sources) are given in Klein et al. (2009). We also include the labeled brain data provided at the MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labeling¹⁷ which we denote as MAL35. This T1-weighted MRI data set consists of 35 subject MRIs taken from the Oasis database¹⁸. The corresponding labels were provided by Neuromorphometrics, Inc¹⁹. under academic subscription.

Comparative evaluation of the two SyN registration approaches was performed within each cohort using a "pseudo-geodesic" approach. Instead of registering every subject to every other subject within a data set, we generated the transforms from each subject to a cohort-specific shape/intensity template. Not only does this reduce the computational time required for finding the pairwise transforms between subjects but prior

Table 1 | Brief overview of the data sets used for the SyN vs. B-spline SyN comparison.

Cohort	Resolution	Number of labels	Preprocessing
CUMC12	[0.86, 0.86, 1.5]	131	Rotated to "cardinal" pose
IBSR18	[1, 1.5, 1]	96	Aligned to talarach, bias corr.
LPBA40	[0.86, 1.5, 0.86]	56	Registered to MNI305, bias corr.
MGH10	[1, 1, 1]	106	Affine-registered to MNI152, bias corr.
MAL35	[1, 1, 1]	138	Average of 3–4 acquisitions, bias corr.

¹⁶http://mindboggle.info/papers/evaluation_NeuroImage2009.php.

¹⁷<https://masi.vuse.vanderbilt.edu/workshop2012>.

¹⁸<http://www.oasis-brains.org>.

¹⁹<http://Neuromorphometrics.com/>.

work has demonstrated improvement in registration with this approach over direct pairwise registration (Klein et al., 2010a). Since the two algorithms have been implemented within the same framework, all registration parameters are identical (i.e., linear registration stage parameters, winsorizing values, etc.) except for the parameters governing the smoothing of the gradient field.

The cohort templates were built using the ANTs script `antsMultivariateTemplateConstruction.sh` which is a multivariate implementation of the work described in Avants et al. (2010). Canonical views for each of the five templates used for this study are given in **Figure 2**. Since calculation of the transform from each subject to the template also includes generation of the corresponding inverse transform, the total transformation from a given subject to any other is determined from the composition of transforms mapping through the template. An example illustration of the geodesic approach is given in **Figure 3**.

Additionally, we refined the labelings for each subject of each cohort using the multi-atlas label fusion algorithm (MALF) developed by Wang et al. (2013) which is also distributed with ANTs. For a given subject within a data set, every other subject was mapped to that subject using the pseudo-geodesic transform. The set of transformed labelings were then used to determine a consensus labeling for that subject. This was to minimize the obvious *observer dimensionality artifacts* where manual raters observe and label in a single dimension at a time. This is most easily seen in the axial or sagittal views of the different cohorts as labelings were done primarily in the coronal view (see **Figure 4**). We include both sets of results. This provides two sets of labels per subject for evaluation²⁰.

²⁰The MALF labels are available at <http://figshare.com/account/projects/196>.

3. RESULTS

As mentioned previously, a template was constructed for each data set (cf **Figure 2**) from all cohort images. Subsequently, each image was registered to its corresponding template using either SyN or B-spline SyN as described previously (prior linear registration stages were identical between the two algorithms). As a brute-force parameter exploration is not a part of this work, we rely on previously reported research (Klein et al., 2009; Avants et al., 2011) and our own experience as authors/developers of the algorithm/software to select parameters which demonstrate robust performance across data sets. For both algorithms, the gradient step was 0.1 for each of the four multi-resolution levels with shrink factors of {6, 4, 2, 1} and Gaussian smoothing for each of those levels being $\mathcal{N}(0, \{9, 4, 1, 0\})$ in terms of voxels. The number of iterations per level were {100, 100, 70, 20} with a convergence threshold of 10^{-9} and window size of 15 iterations²¹.

All processing was performed using the linux cluster at the University of Virginia²² using the PBS Pro queuing system for managing resources. The perl scripts used to create the jobs for the cluster are included in the github account associated with this evaluation. For the data sets used in this study, times for B-spline SyN were approximately 15–40% greater than Gaussian-based SyN using single-threading and a dense metric gradient sampling. Timing data for specific data sets are

²¹As part of the ITKv4 refactoring, we developed several convergence monitoring classes (cf the base class `itk::ConvergenceMonitoringFunction`). Instead of testing for convergence between two successive time points, we use a windowed monitoring function which keeps track of a sequence, or window, of energy values and determines the convergence from the slope of a line fitted to the series.

²²<http://www.uvace.virginia.edu>.

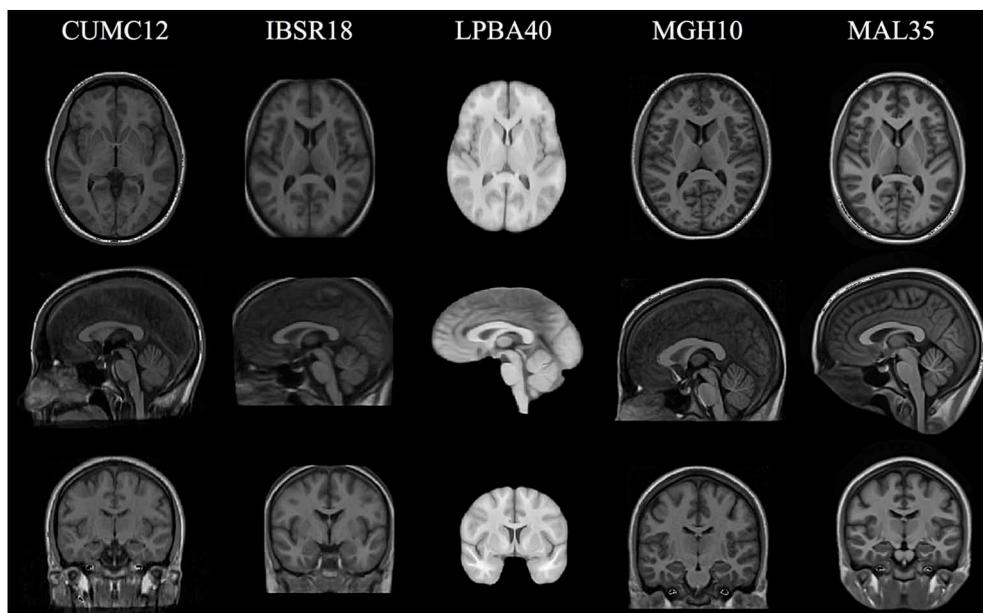


FIGURE 2 | Canonical views for each of the five cohort-specific templates generated using the ANTs tools as described in Avants et al. (2010). The pseudo-geodesic transform between subjects is created from the composition of transforms to/from the relevant template.

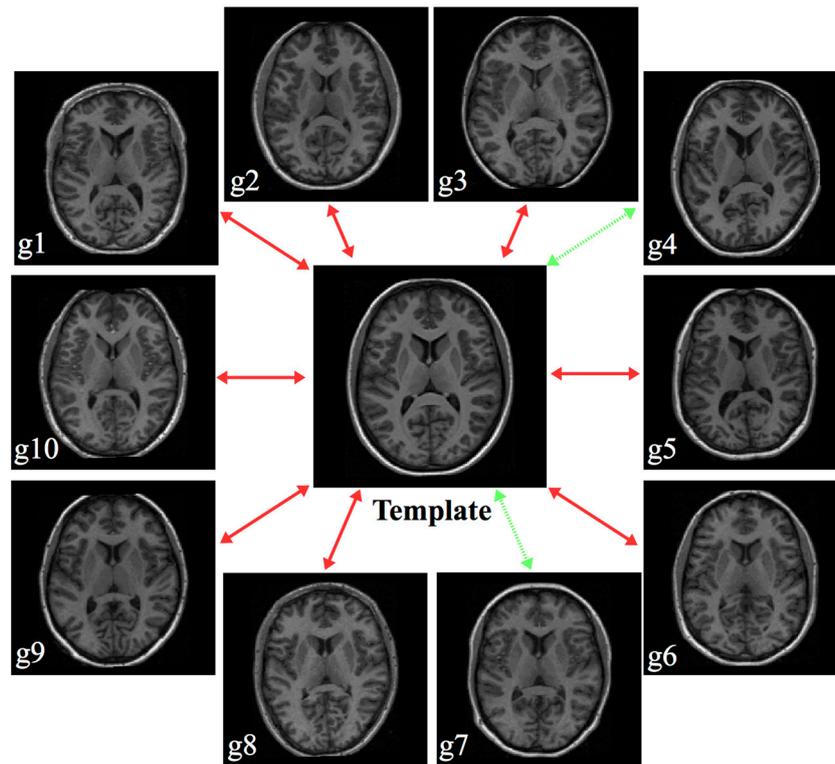


FIGURE 3 | Illustration of generating a pseudo-geodesic for any two subjects within the MGH10 cohort. Once the transforms between the template and each subject are calculated, the mapping between any two subjects is found by composition of forward and inverse transforms. For

example, in the MGH data set, the pseudo-geodesic transform to map Subject g4 to Subject g7 is found by composing the forward transform from g4 to the template with the inverse transform from the template to g7 (green dashed lines).

given in **Table 2** Timing includes both rigid and affine transform optimizations.

The only difference between the two registration settings consists of the Gaussian and B-spline parameters governing the update field smoothing, S_v . In our experience, smoothing of the total field did not improve the results, at least for these data (which conforms with our experience with other data), so the total field smoothing, S_ϕ is 0 for both registration approaches. Specifically, the chosen parameters for the SyN algorithm were: $S_\phi = \mathcal{N}(0, 0)$ and $S_v = \mathcal{N}(0, 3)$ in voxel terms²³. Although our experience with B-spline SyN is much more limited, we were able to choose comparable parameters based on a knot spacing for the update field of 26 mm at the base level which is reduced by a factor of two for each succeeding multiresolution level. This yields a final knot spacing of 3.25 mm²⁴. For comparison, after selecting these smoothing parameters we discovered in the supplementary material of Klein et al. (2009) the similarity to the gradient smoothing parameter for the IRTK FFD algorithm which also used four multi-resolution levels with an initial knot spacing of 20 mm per dimension for a final knot spacing of 2.5 mm.

²³Or, in equivalent antsRegistration command line parlance, -t SyN[0.1, 3, 0].

²⁴Or, equivalently, -t BSplineSyN[0.1, 26, 0].

Quality of overlap using the Dice similarity metric was determined from the transformed labels using the open source ITK implementation described in Tustison and Gee (2009). Both the original labels and MALF labels were warped to the fixed image for comparison using nearest neighbor interpolation. A joint Dice metric value was calculated from the combined labels for each cohort for each of the two SyN methods. These values are rendered in notched box plot format in **Figure 5**. Non-overlapping notches indicate approximately statistically significantly different median values at the 95% confidence level (McGill et al., 1978). For all data sets, the B-spline SyN variant showed a small but statistically significant improvement in overall Dice values. In order to provide a more complete picture of performance differences, we also accounted for label volumetric considerations (Rohlfing, 2012). In **Figure 5** we plotted the Dice value difference between each SyN variant (B-spline SyN—SyN) for each label of each intra-subject registration pair within a data set vs. the volume of the label in the fixed image. Values above and below the dashed line indicate better regional performance for B-spline SyN and SyN, respectively.

To further characterize the deformable transform differences, we calculated the log of the Jacobian determinant of the transformations from each subject to the template and tabulated statistical information within the brain region only. A noticeable difference between the two algorithms was the respective range of

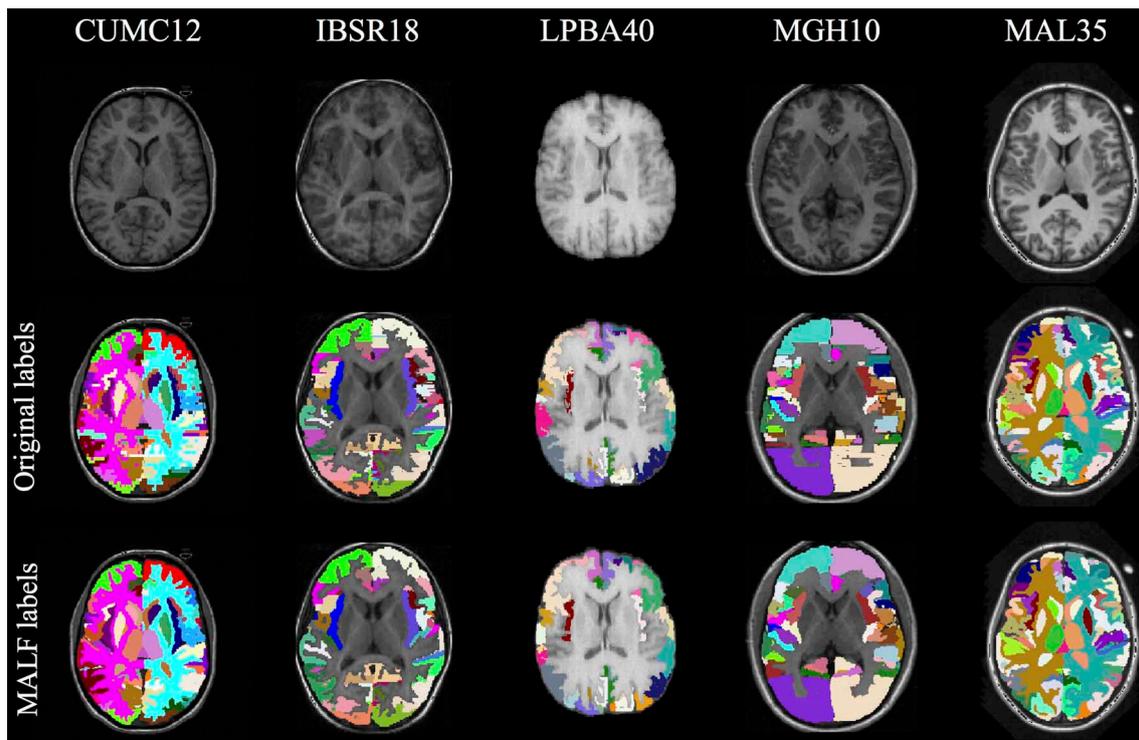


FIGURE 4 | Axial views of sample labelings for a member of each data set. The second row consists of the original labelings with the third row being refined versions of those labelings using the MALF

algorithm (Wang et al., 2013). These refinements provide more consistency between labelings and improved comparative assessments between algorithms.

Table 2 | Timing (in hours) per registration for both SyN algorithms across data sets.

Cohort	SyN	B-spline SyN
CUMC12	8.6 ± 2.3	9.6 ± 2.1
IBSR18	8.3 ± 2.7	10.7 ± 2.8
LPBA40	6.6 ± 0.7	9.5 ± 1.4
MGH10	4.7 ± 1.5	5.8 ± 1.2
MAL35	10.6 ± 1.9	14.2 ± 2.0

values in log Jacobians. We plotted the (95th%—5th%) for each algorithm across all data sets in **Figure 6**. It is apparent that B-spline SyN results exhibit a much greater range of deformation. Qualitative differences are shown in **Figure 7**. In order to ensure randomness to minimize presentation bias in illustrating qualitative results (and given the relative poor performance in humans as random number generators Wagenaar, 1972), we used R to generate uniform random numbers for both subject and axial slice selection. We then used the log Jacobian images to locate regions of maximal difference between the SyN and B-spline SyN results. From **Figure 7** it is quite apparent that the results are very similar which is to be expected considering the almost identical algorithmic make-up between the two approaches (e.g., similarity metric, implementation, linear transforms). However, there are subtle differences particularly in the cortex which help explain both the relative difference in Jacobian and Dice distribution.

4. DISCUSSION AND CONCLUSIONS

B-spline SyN produced slightly greater Dice values than the original SyN. Although actual differences are relatively small, they are statistically significant. By implementing both algorithms in the same code base, we are not only able to eliminate non-regularization components of the registration but we are also able to eliminate implementation differences. Thus, performance disparity can be isolated to smoothing choice. However, even with this restricted focus there are various reasons for the evaluation outcome. These include the approximation-vs.-convolution distinction mentioned earlier for the two regularization approaches (which could also explain the reason why the range of log Jacobian values tend to be significantly higher for B-spline SyN). Also, the fact that the regularization for B-spline SyN is theoretically continuous whereas the truncated Gaussian convolution is only a discrete approximation could be a potential factor.

Additional observations of interest concern the differences in results between the MALF and original labelings for all data sets. Not only was there an overall increase in performance for both algorithms with the MALF labels for all cohorts, but there was also an increase in performance disparity relative to the variance in the resulting Dice values. A possible explanation for this, and one that seems quite plausible, is that the MALF labelings are derived from registering a cohort to the target subject and then performing a consensus labeling (Wang et al., 2013). Both these steps are heavily reliant on image intensity information (in fact, both

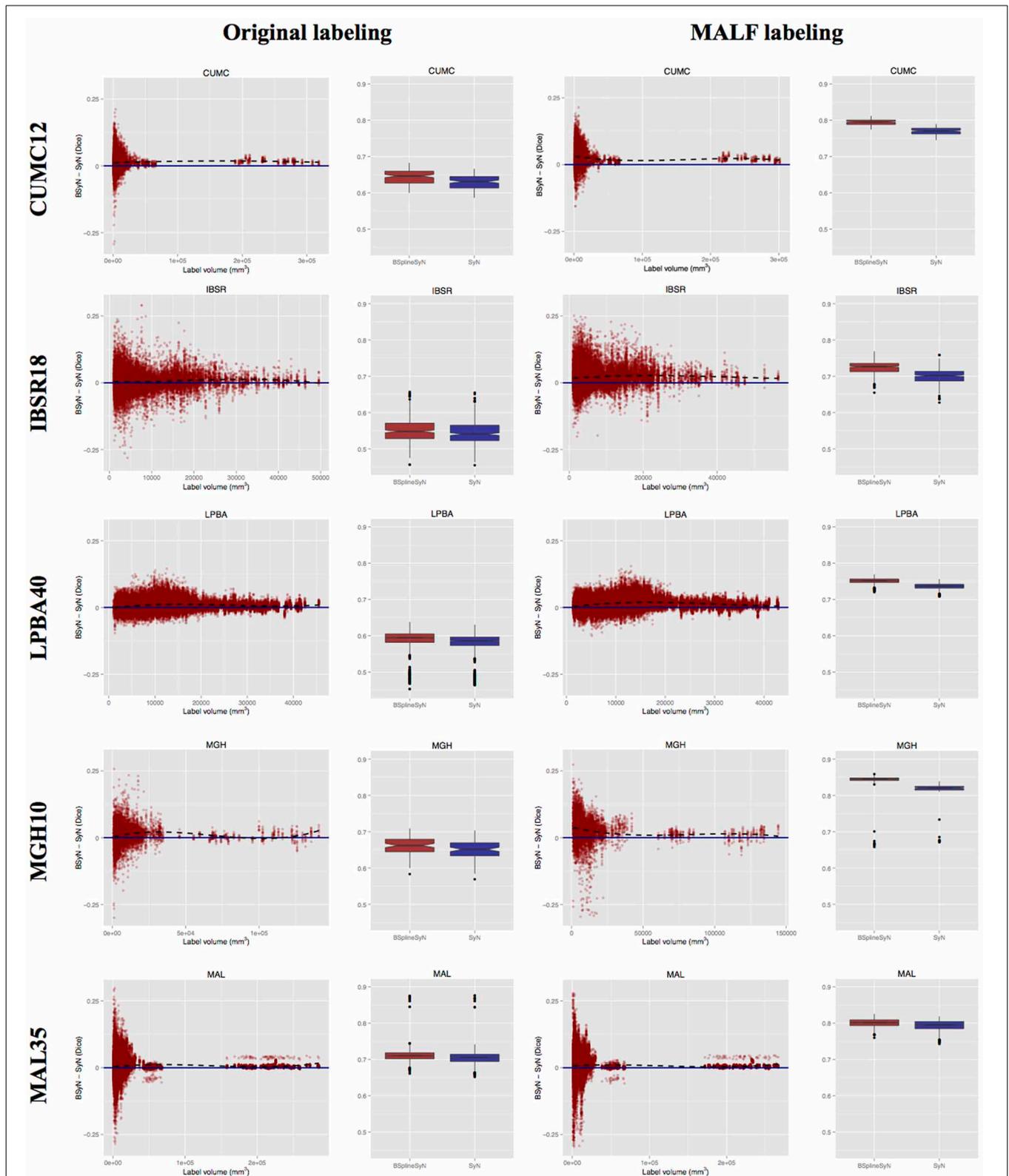
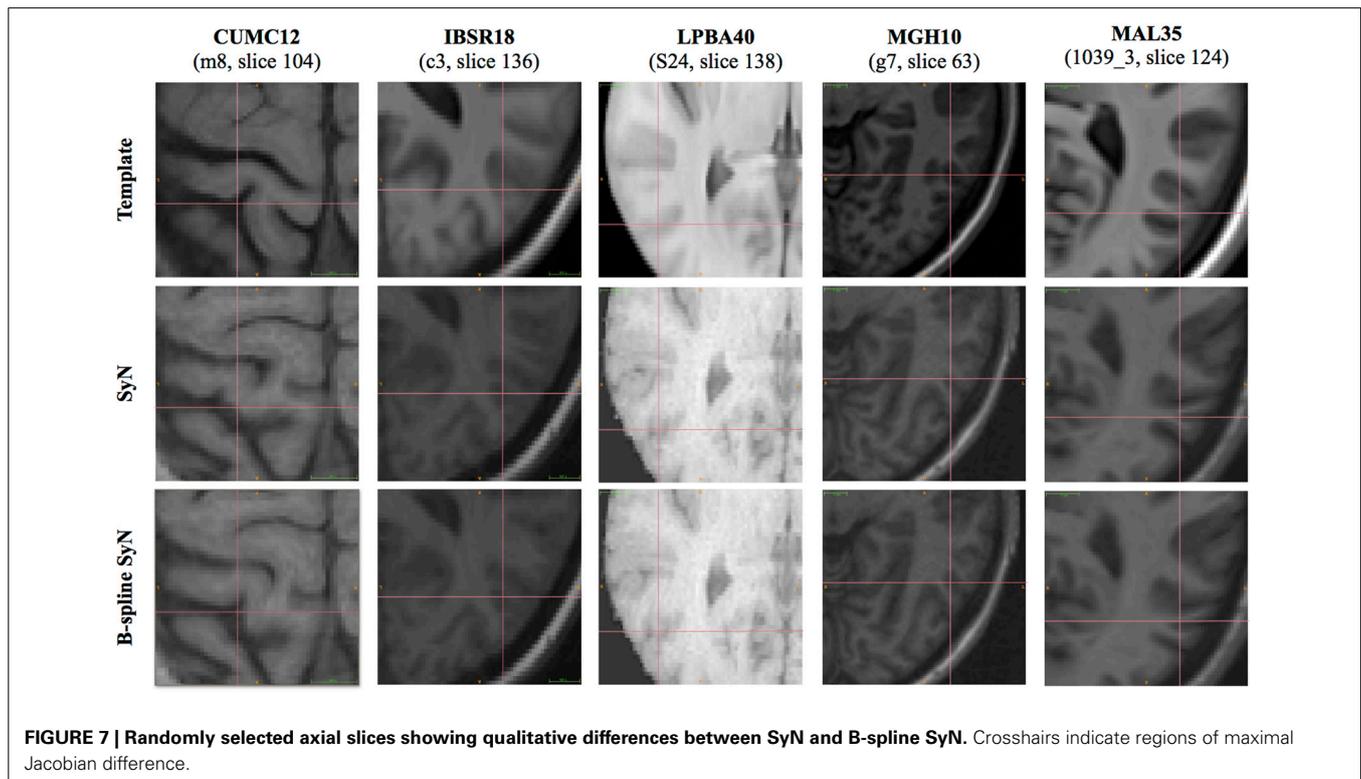
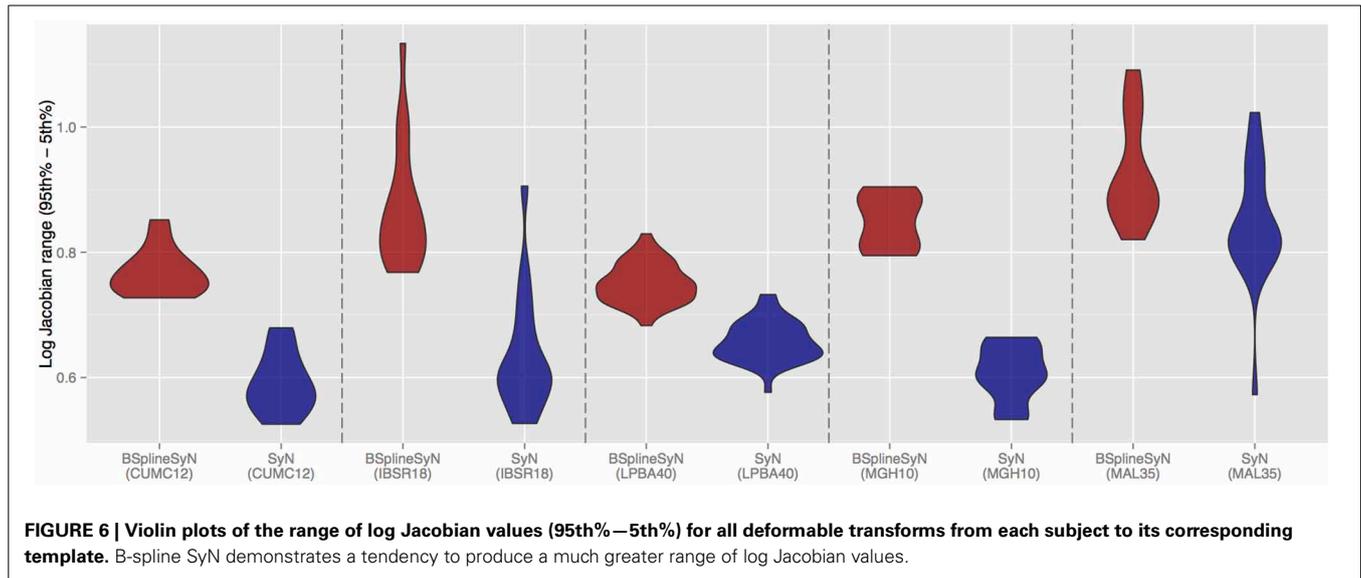


FIGURE 5 | Dice results for both algorithms for each subject warped to every other subject using the pseudo-geodesic transform. Each row corresponds to one of the five data sets used for evaluation. For each data set we include a plotting of all individual label Dice results by volume and a

combined label box plotting. The **left** and **right** halves show the respective results for original and MALF-derived labelings. The black dashed regression line ($y \sim x^3$) illustrates how performance difference varies with label volume for each cohort.



use a form of correlation as the similarity measure for optimization). Since the MALF “correction” tends to group labeled regions according to the same metric used for establishing anatomical correspondences, alignment of these labeled regions seems much more likely which would result in higher Dice metrics. A related effect for labeled data in general is being currently investigated by the authors. From the label volume vs. Dice difference plots, there is no immediately discernible pattern of performance variation with region size. However, a possible confound is that region

definitions vary between cohorts. Although we did not look at region vs. performance difference variation within cohorts, such inquiry is certainly possible as we have made the resulting csv files available with the github repository associated with this work.

Relative to Klein’s study (Klein et al., 2009), it should be noted that the SyN implementation used in that study is found in the ANTS program which is a precursor of the `antsRegistration` program described earlier. Although the theoretical aspects are the same, there are substantial

implementation differences between the two programs. In addition, several parameters varied between the two studies which translated into a more aggressive metric and gradient step in addition to fewer levels and iterations²⁵. In this study we took a more conservative approach based on our continued development and experience resulting in parameters which have proven useful in our cortical thickness pipeline (encapsulated in the ANTs script `antsCorticalThickness.sh`) and our experience with the MICCAI 2013 SATA challenge data. We also ran our own internal experiments with Gaussian-based SyN (both ANTs and `antsRegistration`) using the same conservative parameters on the MAL35 data for which the latter demonstrated slightly improved performance over the former.

Despite the thorough evaluation with multiple data sets, we readily acknowledge the limitations of this study including a very focused application, i.e., healthy brains of a single modality, and absence of a thorough exploration of parameter selection and sensitivity. Although such work might be beneficial (e.g., by aiding other researchers in parameter selection), characterizing parameter permutations of potential interest would expand the current work far beyond its intended scope. However, in addition to this work, SyN has also been previously evaluated in Klein et al. (2009) and Avants et al. (2011) and, based on additional experience and application, the parameter set was modified each time but still yielded excellent performance providing evidence for flexibility in parameter selection. Outside of a range of parameters based on sound engineering principles, experience, and intimate knowledge of both the corresponding algorithms and software, determination of optimal generic parameters even for a specific application is difficult. In fact, the “No Free Lunch Theorem” Wolpert and Macready (1997) emphasizes the importance of prior knowledge in tuning optimization algorithms for a particular application.

One of the advantages that has not been explored in this work is the use of B-spline SyN for small deformation estimation problems such as in pulmonary or cardiac applications. Such problems typically require greater regularization which implies larger discrete kernels for Gaussian convolution. A related issue concerns applications involving severely anisotropic data where the continuous nature of the DMFFD approach might help over Gaussian convolution. Also, we emphasize that underlying the DMFFD approaches is a fitting routine for sparse and scattered data which offers added flexibility over smoothing using discrete convolution where the latter implies regularly placed data on a rectilinear grid for conventional implementations. This advantage could translate into faster running times if only select points are used to drive the registration or make possible more complex registration scenarios involving data arranged continuously within a finite domain (e.g., Tustison et al., 2011). Finally, the possibility of varying data confidence values, as introduced in Tustison and Gee (2006) with DMFFD-based routines would permit incorporation of spatial

preferential weighting (i.e., additional prior knowledge) during optimization. Ongoing work will continue to explore these issues.

A significant amount of research has been devoted to image registration algorithmic development. Given their many salient characteristics particularly with respect to large deformation estimation constrained by topological continuity, diffeomorphic registration approaches have been a particular focus in the neuroimaging community. However, many groups continue to find success with non-diffeomorphic FFD methods (e.g., Rueckert et al., 1999; Klein et al., 2010b). Using our DMFFD framework, B-spline regularization is easily adapted into the diffeomorphic registration framework and performs well compared to analogous algorithms which we demonstrated in this work for the case of the widely-used SyN.

REFERENCES

- Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. (2006). Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM J. Matrix Anal. Appl.* 29, 328–347. doi: 10.1137/050637996
- Ashburner, J. (2007). A fast diffeomorphic image registration algorithm. *Neuroimage* 38, 95–113. doi: 10.1016/j.neuroimage.2007.07.007
- Avants, B. B., Epstein, C. L., Grossman, M., and Gee, J. C. (2008). Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Med. Image Anal.* 12, 26–41. doi: 10.1016/j.media.2007.06.004
- Avants, B. B., Tustison, N. J., Song, G., Cook, P. A., Klein, A., and Gee, J. C. (2011). A reproducible evaluation of ants similarity metric performance in brain image registration. *Neuroimage* 54, 2033–2044. doi: 10.1016/j.neuroimage.2010.09.025
- Avants, B. B., Yushkevich, P., Pluta, J., Minkoff, D., Korczykowski, M., Detre, J., et al. (2010). The optimal template effect in hippocampus studies of diseased populations. *Neuroimage* 49, 2457–2466. doi: 10.1016/j.neuroimage.2009.09.062
- Bajcsy, R., and Kovacic, S. (1989). Multiresolution elastic matching. *Comput. Vis. Graph. Image Process.* 46, 1–21. doi: 10.1016/S0734-189X(89)80014-3
- Beg, M. F., Miller, M. I., Trounev, A., and Younes, L. (2005). Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *Int. J. Comput. Vis.* 61, 139–157. doi: 10.1023/B:VISI.0000043755.93987.a
- Bouma, H., Vilanova, A., Bescos, J. O., Ter Haar Romeny, B. M., and Gerritsen, F. A. (2007). “Fast and accurate Gaussian derivatives based on B-splines,” in *Scale Space and Variational Methods in Computer Vision. Lecture Notes in Computer Science*. Vol. 4485, eds F. Sgallari, A. Murli, and N. Paragios (Berlin: Springer), 406–417. doi: 10.1007/978-3-540-72823-8_35
- Bro-Nielsen, M., and Gramkow, C. (1996). “Fast fluid registration of medical images,” in *Proceedings of the 4th International Conference on Visualization in Biomedical Computing, VBC '96* (London, UK: Springer-Verlag), 267–276.
- Christensen, G. E., Rabbitt, R. D., and Miller, M. I. (1996). Deformable templates using large deformation kinematics. *IEEE Trans. Image Process.* 5, 1435–1447. doi: 10.1109/83.536892
- De Craene, M., Piella, G., Camara, O., Duchateau, N., Silva, E., Doltra, A., et al. (2011). Temporal diffeomorphic free-form deformation: application to motion and strain estimation from 3d echocardiography. *Med. Image Anal.* 16, 427–450. doi: 10.1016/j.media.2011.10.006
- Dupuis, P., and Grenander, U. (1998). Variational problems on flows of diffeomorphisms for image matching. *Q. Appl. Math.* LVI, 587–600.
- Gee, J. C., Reivich, M., and Bajcsy, R. (1993). Elastically deforming 3D atlas to match anatomical brain images. *J. Comput. Assist. Tomogr.* 17, 225–236. doi: 10.1097/00004728-199303000-00011
- Hsu, W. M., Hughes, J. F., and Kaufman, H. (1992). “Direct manipulation of free-form deformations,” in *SIGGRAPH*, ed J. J. Thomas (New York, NY: ACM), 177–184.
- Ibanez, L., Schroeder, W., Ng, L., and Cates, J. (2005). *The ITK Software Guide*. 2nd Edn. Kitware, Inc. ISBN 1-930934-15-7. Available online at: <http://www.itk.org/ItkSoftwareGuide.pdf>.
- Klein, A., Andersson, J., Ardekani, B. A., Ashburner, J., Avants, B., Chiang, M.-C., et al. (2009). Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. *Neuroimage* 46, 786–802. doi: 10.1016/j.neuroimage.2008.12.037

²⁵Specifically, as provided in the supplementary material of Klein et al. (2009), the command line parameters were: ANTS 3 -m PR[$\{\text{fixed}\}$, $\{\text{moving}\}$,1,2] -o $\{\text{output}\}$ -r Gauss[2,0] -t SyN[0.5] -i 30x99x11 -use-Histogram-Matching.

- Klein, A., Ghosh, S. S., Avants, B., Yeo, B. T. T., Fischl, B., Ardekani, B., et al. (2010a). Evaluation of volume-based and surface-based brain image registration methods. *Neuroimage* 51, 214–220. doi: 10.1016/j.neuroimage.2010.01.091
- Klein, S., Staring, M., Murphy, K., Viergever, M. A., and Pluim, J. P. W. (2010b). Elastix: a toolbox for intensity-based medical image registration. *IEEE Trans. Med. Imaging* 29, 196–205. doi: 10.1109/TMI.2009.2035616
- Lee, S., Wolberg, G., and Shin, S. (1997). Scattered data interpolation with multilevel B-splines. *IEEE Trans. Visual. Comput. Graph.* 3, 228–244. doi: 10.1109/2945.620490
- Mattes, D., Haynor, D. R., Vesselle, H., Lewellen, T. K., and Eubank, W. (2003). PET-CT image registration in the chest using free-form deformations. *IEEE Trans. Med. Imaging* 22, 120–128. doi: 10.1109/TMI.2003.809072
- McGill, R., Tukey, J., and Larsen, W. (1978). Variations of box plots. *Am. Statist.* 32, 12–16. doi: 10.1080/00031305.1978.10479236
- Modat, M., Ridgway, G. R., Daga, P., Cardoso, M. J., and Hawkes, D. J. (2011). “Log-Euclidean free-form deformation,” in *Progress in Biomedical Optics and Imaging—Proceedings of SPIE*. Vol. 7962 (Lake Buena Vista, FL).
- Modat, M., Ridgway, G. R., Taylor, Z. A., Lehmann, M., Barnes, J., Hawkes, D. J., et al. (2010). Fast free-form deformation using graphics processing units. *Comput. Methods Program. Biomed.* 98, 278–284. doi: 10.1016/j.cmpb.2009.09.002
- Murphy, K., van Ginneken, B., Reinhardt, J. M., Kabus, S., Ding, K., Deng, X., et al. (2011). Evaluation of registration methods on thoracic CT: the EMPIRE10 challenge. *IEEE Trans. Med. Imaging* 30, 1901–1920. doi: 10.1109/TMI.2011.2158349
- Pennec, X., Cachier, P., and Ayache, N. (1999). “Understanding the “Demon’s algorithm”: 3D non-rigid registration by gradient descent,” in *Proceedings of the Second International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI '99* (London, UK: Springer-Verlag) 597–605.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing*. 3rd Edn. New York, NY: Cambridge University Press.
- Risser, L., Vialard, F.-X., Wolz, R., Murgasova, M., Holm, D. D., Rueckert, D., et al. (2011). Simultaneous multi-scale registration using large deformation diffeomorphic metric mapping. *IEEE Trans. Med. Imaging* 30, 1746–1759. doi: 10.1109/TMI.2011.2146787
- Rohlfing, T. (2012). Image similarity and tissue overlaps as surrogates for image registration accuracy: widely used but unreliable. *IEEE Trans. Med. Imaging* 31, 153–163. doi: 10.1109/TMI.2011.2163944
- Rohlfing, T., Maurer, C. R., Jr., Bluemke, D. A., and Jacobs, M. A. (2003). Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint. *IEEE Trans. Med. Imaging* 22, 730–741. doi: 10.1109/TMI.2003.814791
- Rueckert, D., Aljabar, P., Heckemann, R. A., Hajnal, J. V., and Hammers, A. (2006). Diffeomorphic registration using B-splines. *Med. Image Comput. Comput. Assist. Interv.* 9(Pt 2), 702–709. doi: 10.1007/11866763_86
- Rueckert, D., Sonoda, L. I., Hayes, C., Hill, D. L., Leach, M. O., and Hawkes, D. J. (1999). Nonrigid registration using free-form deformations: application to breast MR images. *IEEE Trans. Med. Imaging* 18, 712–721. doi: 10.1109/42.796284
- Sederberg, T. W., and Parry, S. R. (1986). Free-form deformation of solid geometric models. *SIGGRAPH Comput. Graph.* 20, 151–160. doi: 10.1145/15886.15903
- Shackelford, J. A., Kandasamy, N., and Sharp, G. C. (2010). On developing B-spline registration algorithms for multi-core processors. *Phys. Med. Biol.* 55, 6329–6351. doi: 10.1088/0031-9155/55/21/001
- Szeliski, R., and Coughlan, J. (1997). Spline-based image registration. *Int. J. Comput. Vis.* 22, 199–218. doi: 10.1023/A:1007996332012
- Thévenaz, P., Ruttimann, U. E., and Unser, M. (1998). A pyramid approach to sub-pixel registration based on intensity. *IEEE Trans. Image Process* 7, 27–41. doi: 10.1109/83.650848
- Thirion, J. P. (1998). Image matching as a diffusion process: an analogy with Maxwell’s demons. *Med. Image Anal.* 2, 243–260. doi: 10.1016/S1361-8415(98)80022-4
- Tustison, N. J., and Avants, B. B. (2012a). “Diffeomorphic directly manipulated free-form deformation image registration via vector field flows,” in *WBIR. Lecture Notes in Computer Science*. Vol. 7359, eds B. M. Dawant, G. E. Christensen, J. M. Fitzpatrick, and D. Rueckert (Berlin: Springer), 31–39. doi: 10.1007/978-3-642-31340-0_4
- Tustison, N. J., and Avants, B. B. (2012b). The TVDMFFDVR algorithm. *Insight J.* 1–10. Available online at: <http://hdl.handle.net/10380/3334>
- Tustison, N. J., Avants, B. B., and Gee, J. C. (2009). Directly manipulated free-form deformation image registration. *IEEE Trans. Image Process* 18, 624–635. doi: 10.1109/TIP.2008.2010072
- Tustison, N. J., Awate, S. P., Song, G., Cook, T. S., and Gee, J. C. (2011). Point set registration using havrda-charvat-tsallis entropy measures. *IEEE Trans. Med. Imaging* 30, 451–460. doi: 10.1109/TMI.2010.2086065
- Tustison, N. J., and Gee, J. C. (2006). “Generalized n -d C^k B-spline scattered data approximation with confidence values,” in *Proceedings of the Third international conference on Medical Imaging and Augmented Reality, Miar’06*, (Berlin: Springer-Verlag), 76–83. doi: 10.1007/11812715_10
- Tustison, N. J., and Gee, J. C. (2009). Introducing Dice, Jaccard, and other label overlap measures to ITK. *Insight J.* 1–4. Available online at: <http://hdl.handle.net/10380/3141>
- Tustison, N. J., Johnson, H. J., Rohlfing, T., Klein, A., Ghosh, S. S., Ibanez, L., et al. (2013). Instrumentation bias in the use and evaluation of scientific software: recommendations for reproducible practices in the computational sciences. *Front. Neurosci.* 7:162. doi: 10.3389/fnins.2013.00162
- Wagenaar, W. A. (1972). Generation of random sequences by human subjects: a critical survey of the literature. *Psychol. Bull.* 77, 65–72. doi: 10.1037/h0032060
- Wang, H., Suh, J. W., Das, S. R., Pluta, J., Craige, C., and Yushkevich, P. A. (2013). Multi-atlas segmentation with joint label fusion. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 611–623. doi: 10.1109/TPAMI.2012.143
- Wolpert, D., and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* 1, 67–82. doi: 10.1109/4235.585893

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Received: 28 October 2013; paper pending published: 16 November 2013; accepted: 08 December 2013; published online: 23 December 2013.

Citation: Tustison NJ and Avants BB (2013) Explicit B-spline regularization in diffeomorphic image registration. *Front. Neuroinform.* 7:39. doi: 10.3389/fninf.2013.00039 This article was submitted to the journal *Frontiers in Neuroinformatics*.

Copyright © 2013 Tustison and Avants. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.