A Symmetric Prior for the Regularisation of Elastic Deformations: Improved Anatomical Plausibility in Nonlinear Image Registration

Frederik J Lange[a,*], John Ashburner[b], Stephen M Smith[a], Jesper L R Andersson[a]

[a] *Centre for Functional MRI of the Brain (FMRIB), Wellcome Centre for Integrative Neuroimaging, Nuffield Department of Clinical Neurosciences, University of Oxford, John Radcliffe Hospital, Headley Way, Oxford, OX3 9DU, UK*

[b] *Wellcome Centre for Human Neuroimaging, UCL Institute of Neurology, University College London, 12 Queen Square, London, WC1N 3BG, UK*

Abstract

Nonlinear registration is an important tool in Neuroimaging. It facilitates averaging and comparisons across multiple subjects, as well as reporting of data in a common anatomical frame of reference. It is, however, a fundamentally ill-posed problem, with many possible solutions which minimise a given dissimilarity metric equally well. We present a novel regularisation method which aims to selectively drive solutions towards those which would be considered *anatomically plausible* by penalising unlikely lineal, areal and volumetric deformations. Additionally our penalty is symmetric in the sense that geometric expansions and contractions are penalised equally which encourages inverse-consistency. We show that our method is able to match the registration quality, as measured by overlap scores of labelled cortical regions, of state of the art elastic (FNIRT) and plastic (ANTs) registration frameworks. Crucially, this is achieved with a significant reduction in how aggressively the images are deformed, as gauged by both the magnitude and spatial distribution of Jacobian determinants. Finally, extensive use of GPU parallelisation has allowed us to implement what is a highly computationally

*Corresponding Author

Email addresses: `frederik.lange@dtc.ox.ac.uk` (Frederik J Lange), `j.ashburner@ucl.ac.uk` (John Ashburner), `steve@fmrib.ox.ac.uk` (Stephen M Smith), `jesper.andersson@ndcn.ox.ac.uk` (Jesper L R Andersson)

intensive optimisation strategy whilst maintaining reasonable run times, on the order of half an hour.

Keywords: Nonlinear Registration, Elastic Deformation, Regularization, GPU Parallelization, Anatomical Plausibility, Symmetry

1. Introduction

Non-linear registration is used in neuroimaging to deform images of individual brains into some common space (normalising both size and shape). Often the registration is based on structural, e.g., T_1-weighted, images. The resulting warp is subsequently applied to both structural images as well as images depicting some function of interest such as BOLD or diffusion-derived connectivity. The rationale behind this is typically to facilitate statistical analysis of those functional data across subjects and populations.

An important distinction in this context is between volume- and surface-based methods. The former attempt to find the inter-subject mappings in the original (3D) space. The latter, in contrast, are typically only concerned with the cortex and attempt to inflate the folded 2D-manifold (the cortex) onto a sphere before finding the (2D) inter-subject mapping on that sphere (e.g., Fischl (2012); Robinson et al. (2014)). In this paper we suggest a novel volume based method.

Existing volume based methods can broadly be categorised based on:

Construction of warps As part of the iterative process a warp can be updated by adding an update, or by warping the previous warps by an update. The former is often referred to as an *elastic* or a *small deformation* framework, as opposed to a *plastic* or *large deformation* framework for the latter (Miller et al., 2003). Note that for infinitesimally small displacements, the concept of composing warps can be extended to the integration of a velocity field (Beg et al., 2005).

Regularisation of warps Warps can be *regularised*, i.e., have smoothness imposed on them, by adding a penalty term to the cost-function, or by explicitly smoothing the updates.

Similarity measure This is a scalar measure that assesses the similarity/difference between the images and whose maximisation/minimisation drives the registration.

The present paper is concerned mainly with the second of these, the regularisation, though we will touch upon the construction of warps and the effect of similarity metric choice as well.

Historically, for the small deformation framework, one important aspect of the regularisation was to try to ensure that the warps stayed one-to-one and onto i.e., that each point in image A mapped to a unique point in image B, and that each point in image B could be reached from image A. This also means that the mapping is invertible. This was often attempted by adding a penalty term to the cost-function such that one simultaneously minimised the difference between the images (dissimilarity) and some function of the warps (regularisation). That function would often be borrowed from mechanics, such as bending energy (Bookstein, 1997), membrane energy (Amit et al., 1991) or linear-elastic energy (Miller et al., 1993) (whose formulations are summarised in Ashburner and Friston (1999)). However, none of those functions had any specifically biological relevance to the problem of mapping individual brains to each other. Moreover, they do not explicitly guarantee invertibility, so it was common practice to empirically calibrate the relative weights of the image difference and the warp penalty so that the resulting warps were (almost) always invertible. But that typically meant a quite high weight for the penalty term which resulted in an inability to properly model large deformations.

This limitation led to the development of the large deformation framework where warps are updated by warping (resampling) the previous warp by an update warp (equivalent to a composition of multiple small warps). If both the previous and the update warps are invertible, this guarantees that the composed warp is also invertible. Therefore, as long as one ensures that each update warp is invertible the end result will be too. Furthermore, certain methods (such as LDDMM (Beg et al., 2005) and Geodesic Shooting (Ashburner and Friston, 2011)) seek solutions where the overall set of composed warps (or integrated velocity field) are as smooth as possible, rather than greedily smoothing at each step. The result is then that these methods are able to model large deformations while remaining invertible by taking many small update steps, leading to better matching than could be achieved within the small deformation framework.

However, invertibility is not a panacea for all problems associated with inter-subject mapping of brains and is no more informed by the actual problem at hand than were the various penalty functions borrowed from mechanics. It can, and frequently does, lead to warps that correspond to implausibly

large local changes in volume and/or shape (Coalson et al., 2018).

In this paper we suggest using a small deformation framework together with a penalty function that explicitly penalises changes in both volume and shape, and that goes to infinity as relative volumes approach zero (which would break having a one-to-one relationship). We will refer to this penalty as the Symmetric Prior for the Regularisation of Elastic Deformations (SPRED). The mathematical formulation and theoretical basis of the penalty are explained in Section 2.3. It has been used in the past (Ashburner et al., 1999, 2000), but the computational cost is such that at the time it was not practically feasible. With the advent of *general-purpose computing on graphics processing units* (GPGPU) that is no longer the case, and we describe our implementation using NVIDIA's CUDA framework (NVIDIA, 2019) of a registration algorithm based on that penalty function in Section2.4.

The rationale behind the suggested penalty function is twofold,

- To minimise geometric distortions and to ensure the warps are one-to-one and onto.
- To be symmetric in the sense that a geometric (lineal, areal or volumetric) change by a factor of 2 is penalised the same as one by a factor of 0.5.

The first desideratum comes from the notion that brain tissue implements function, be that function integration of signal in the grey matter or transmission of signal in the white matter. The implementation of that function will necessarily occupy some physical space and it would seem unlikely that a given function can be implemented in a very small fraction of space in one brain compared to in another. Similar arguments can be made for many other biological tissues. A penalty should therefore favour no volume changes at all. It should also rapidly increase as the relative volumes become unrealistic, approaching infinity as the volume changes gets close to breaking the one-to-one and onto criteria, thereby never allowing negative relative volumes. But it should also prevent unrealistic shape changes. Given what we know about the structure of both grey and white matter it is hard to picture a situation where a function implemented in a $1\times1\times1\,\mathrm{mm}^3$ cube in one brain might occupy a $0.01\times10\times10\,\mathrm{mm}^3$ sheet or a $0.1\times0.1\times100\,\mathrm{mm}^3$ stick in another. It should therefore also penalise shape changes that we would consider unrealistic. Note that this can all be reduced to penalising linear changes since that would automatically also penalise shape and volume changes.

The second property, symmetry, comes from the desire that a registration algorithm should be inverse consistent, i.e., that the transform that is obtained from registering image A to image B is the inverse of that obtained from registering B to A (Christensen, 1999). An expansion in the forward direction (from A to B) corresponds to a contraction in the reverse (from B to A), and therefore if we wish our transform to be independent of the ordering of our images then these effects should be penalised equally.

2. Methods

2.1. Registration Framework

We now broadly describe the registration framework within which our SPRED penalty is implemented in order that specific details of the implementation are more readily understandable.

SPRED forms part of the development of our new MultiModal Registration Framework (MMORF) tool. MMORF is a volumetric registration tool whose underlying transformation model is a 3D cubic B-spline parametrised free-form deformation. The parameter space over which the optimisation is performed therefore represents the spline coefficients of 3 displacement fields (x, y and z-warps). A mean-squares (MSQ) similarity (or rather *dissimilarity*) metric is the data-consistency term optimised during registration. Regularisation penalties are calculated in the reference image space. A multi-smoothing, multi-resolution approach is employed to encourage convergence towards a globally optimum solution. The preferred optimisation method is the Levenberg-Marquardt variant of Gauss-Newton (Press et al., 2007), however Scaled Conjugate Gradient (Moller, 1993) can also be used when memory limitations are a consideration. From the outset MMORF has been designed to leverage GPU parallelisation, which is a largely why SPRED is computationally tractable within this framework.

2.2. Intuition

In the following section we will detail how to calculate the suggested penalty, and why it is an ideal candidate for Single Instruction Multiple Data (SIMD) style parallelisation on a GPU (NVIDIA, 2019). For the estimation of the warp parameters we use the Gauss-Newton method, so we will also need to calculate the gradient of the penalty with respect to the spline coefficients of the warps and a Gauss-Newton approximation to the Hessian.

We attempt now to give an intuitive outline of the technical/mathematical formalism which follows in Section 2.3. At each voxel one can calculate a Jacobian matrix using the values of the surrounding warp spline-coefficients. The singular values of this matrix represent the orthogonal scaling of that voxel due to the warp. As the SPRED penalty is a function of these singular values, we can then calculate this voxel's contribution to the total penalty. Those calculations are a perfect match for SIMD parallelisation with one computational thread per *voxel*, with each thread performing identical calculations but on different data. The individual voxel contributions are subsequently summed (a *reduction* in GPU parlance) to obtain the total penalty for the warp.

In order to calculate the gradient of the penalty, one needs to apply the chain rule, which says that:

$$\frac{d}{dx}\left[f(g(x))\right] = \frac{df}{dg}\frac{dg}{dx} \tag{1}$$

Remembering that for example g might be a vector-valued function of a vector-valued argument x which would make df/dg a vector and dg/dx a matrix.

In our case the contribution to an element of the gradient from a given voxel is a scalar valued function of the vector of singular values of the local Jacobian matrix. These in turn are each a function of the vector of elements of the Jacobian matrix. Finally, these elements are functions of the vector of spline-coefficients whose support includes that voxel. The first two factors are again calculated using one thread per *voxel*, while the third factor uses one thread per *spline-coefficient*. Similarly each element of the total gradient is calculated by one thread per *spline-coefficient*.

To understand how the Hessian matrix is calculated consider a matrix $\mathbf{A}$ where each row corresponds to a voxel and each column to a warp spline-coefficient, and where $\mathbf{A}_{ij}$ is the rate of change of the penalty contribution from voxel i with respect to coefficient j. The Gauss-Newton approximation to the Hessian then becomes $\mathbf{A}^{\mathrm{T}}\mathbf{A}$. However, for practical reasons, $\mathbf{A}$ is never explicitly calculated or stored and instead the Hessian $\mathbf{H}$ is calculated directly. In this instance we use one thread per non-zero *element of* $\mathbf{H}$ for the final calculation, but there are a number of preceding steps which employ the same one thread per *voxel* approach as in the gradient calculations.

2.3. Theory

We now formalise the concepts introduced in Section 2.2 in terms of the mathematics required for incorporating the SPRED penalty into the optimisation strategy of our registration framework.

2.3.1. Jacobian Matrix

Consider the generic problem of registering some 3D moving image $\mathbf{f}$ to a reference image $\mathbf{g}$, where both are defined on $\mathbb{R}^3$. Let x, y and z be the 3 orthogonal directions in $\mathbb{R}^3$. We define a transformation $\mathbf{W}$ on the domain of $\mathbf{g}$ with parameters $\vec{w}$, such that $\mathbf{f}(\vec{w})$ becomes our transformed moving image. At each point (x, y, z) :

$$\mathbf{W}(x, y, z) = [\mathbf{W}_x, \mathbf{W}_y, \mathbf{W}_z] \tag{2}$$

$$= [x + \mathbf{D}_x, y + \mathbf{D}_y, z + \mathbf{D}_z] \tag{3}$$

Where $\mathbf{D}$ is a displacement field. If $\mathbf{W}$ defines a continuous, differentiable function on $\mathbb{R}^3$, then we may calculate a local Jacobian matrix $\mathbf{J}$ at any point in $\mathbf{g}$, with:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \mathbf{W}_x}{\partial x} & \frac{\partial \mathbf{W}_x}{\partial y} & \frac{\partial \mathbf{W}_x}{\partial z} \\ \frac{\partial \mathbf{W}_y}{\partial x} & \frac{\partial \mathbf{W}_y}{\partial y} & \frac{\partial \mathbf{W}_y}{\partial z} \\ \frac{\partial \mathbf{W}_z}{\partial x} & \frac{\partial \mathbf{W}_z}{\partial y} & \frac{\partial \mathbf{W}_z}{\partial z} \end{bmatrix} \tag{4}$$

$$= \begin{bmatrix} (1 + \frac{\partial \mathbf{D}_x}{\partial x}) & \frac{\partial \mathbf{D}_x}{\partial y} & \frac{\partial \mathbf{D}_x}{\partial z} \\ \frac{\partial \mathbf{D}_y}{\partial x} & (1 + \frac{\partial \mathbf{D}_y}{\partial y}) & \frac{\partial \mathbf{D}_y}{\partial z} \\ \frac{\partial \mathbf{D}_z}{\partial x} & \frac{\partial \mathbf{D}_z}{\partial y} & (1 + \frac{\partial \mathbf{D}_z}{\partial z}) \end{bmatrix} \tag{5}$$

Where ${}^{\partial \mathbf{W}_i}/_{\partial j}$ is the partial derivative of the i component of $\mathbf{W}$ in the j direction.

$\mathbf{J}$ then describes how the image is locally scaled and rotated by the action of $\mathbf{W}$. These actions can be decoupled by applying singular value decomposition (SVD) to $\mathbf{J}$, such that:

$$\mathbf{J} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} \tag{6}$$

$\mathbf{U}$ and $\mathbf{V}$ are unitary matrices describing the rotational effect of $\mathbf{J}$, and will not be considered further as they do not geometrically distort the image in

any way. $\mathbf{S}$ is a diagonal matrix containing the singular values and represents the orthogonal scaling effect of $\mathbf{J}$. It is the individual elements s_i of $\mathbf{S}$ which will be penalised in our regularisation, and we will refer to them as the Jacobian Singular Values (JSVs).

2.3.2. Penalty Function

The penalty is based on the prior belief that the JSVs are drawn from a lognormal distribution. Such a penalty meets the biological plausibility arguments set out in Section 1 as:

- $s_i = 1$ is most likely
- $s_i = 0$ and $s_i = \infty$ are infinitely unlikely
- $s_i = a$ and $s_i = \frac{1}{a}$ are equally likely

Now, we define $C(\vec{w})$ as the penalty (or *cost*) associated with with $\mathbf{W}$:

$$C(\vec{w}) = \sum_{n=1}^{N} c_n(\vec{w}) \tag{7}$$

$$= \sum_{n=1}^{N} v\left(1 + |\mathbf{J}_n|\right) \sum_{i=1}^{3} \log^2 s_i \tag{8}$$

Where:

$$\begin{aligned}
C &= \text{total cost/penalty} \\
\vec{w} &= \text{transformation parameters} \\
N &= \text{total voxels in image} \\
c_n &= \text{cost/penalty at voxel } n \\
v &= \text{voxel volume} \\
\mathbf{J}_n &= \mathbf{J}_n(\vec{w}) = \text{Jacobian matrix at voxel } n \\
s_i &= s_i(\vec{w}) = i^{\text{th}} \text{ singular value of } \mathbf{J}_n
\end{aligned}$$

From Equation 8 we recognise that the $\log^2 s_i$ term represents our lognormal prior on the JSVs. However this prior is only based on the distribution of singular values in $\mathbf{g}$. Therefore we have also included a term $v(1 + |\mathbf{J}n|)$ to ensure that our penalty is truly symmetric in $\mathbf{f}$ by accounting for the total volume in both images being penalised.

In practice, computation of the logarithm in Equation 7 (and its associated effect on the gradient and Hessian) may add a significant computational overhead (Ashburner et al., 2000). We therefore apply the following approximation:

$$\log^2 x \approx x + \frac{1}{x} - 2 \tag{9}$$

And, recognising that:

$$\log^2 x = \frac{\log^2 (x^2)}{4} \tag{10}$$

and that:

$$\mathrm{tr}(\mathbf{A}^{\mathrm{T}}\mathbf{A}) = \sum_{i=1}^{3} \lambda_i(\mathbf{A}^{\mathrm{T}}\mathbf{A}) = \sum_{i=1}^{3} \left(s_i(\mathbf{A})\right)^2 \tag{11}$$

where $\lambda_i(\mathbf{A}^{\mathrm{T}}\mathbf{A})$ denotes the ith eigenvalue of $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ and where $s_i(\mathbf{A})$ denotes the ith singular value of $\mathbf{A}$. We may then write:

$$C\left(\vec{w}\right) \approx \sum_{n=1}^{N} v\left(1 + |\mathbf{J}_n|\right) \sum_{i=1}^{3} \left(s_i^2 + \frac{1}{s_i^2} - 2\right) \Big/ 4 \tag{12}$$

$$= \sum_{n=1}^{N} v\left(1 + |\mathbf{J}_n|\right) \mathrm{tr}\left(\mathbf{J}_n^{\mathrm{T}}\mathbf{J}_n + \left(\mathbf{J}_n^{-1}\right)^{\mathrm{T}} \mathbf{J}_n^{-1} - 2\mathbf{I}\right) / 4 \tag{13}$$

An important benefit of this approximation is that our cost function is now an explicit function of only the elements of $\mathbf{J}$, and therefore we never need to calculate the actual SVD of $\mathbf{J}$.

2.3.3. Transformation Model

The previous sections have used a generic transformation $\mathbf{W}$, but at this point it becomes necessary to define the actual transformation model used. As in Equation 3, we utilise a displacement field $\mathbf{D}$ where each of the three directional components ($\mathbf{D}_x$, $\mathbf{D}_y$ and $\mathbf{D}_z$) in $\mathbb{R}^3$ is composed of a basis set of uniformly spaced, 3D, cubic B-splines ($\mathbf{B}_x$, $\mathbf{B}_y$ and $\mathbf{B}_z$). Our transformation parameters $\vec{w}$ are then defined to be the coefficients of each of the B-splines in our basis set. If M B-splines are required to fully cover the N voxels in our reference image $\mathbf{g}$, then we require $3M$ parameters to fully describe $\mathbf{D}$ (i.e. M parameters for each displacement direction).

2.3.4. Gradient and Hessian

The penalty function as defined in Section 2.3.2 will be included into a Gauss-Newton style optimisation framework, and therefore we require the gradient and the Gauss-Newton approximation to the Hessian of Equation 12. We note that for a valid application of Gauss-Newton (i.e., for being able to use the Gauss-Newton approximation to the Hessian) the function being minimised must be of the form $y = \frac{1}{2}a(x)^2$, which Equation 13 is not (Chen, 2011). However, by making the substitution:

$$C\left(\vec{w}\right) = \frac{1}{2}\sum_{n=1}^{N} a_n^2\left(\vec{w}\right) = \sum_{n=1}^{N} c_n\left(\vec{w}\right) \tag{14}$$

it becomes of that form. This means that:

$$\nabla C\left(\vec{w}\right) = \sum_{n=1}^{N} \frac{\partial a_n}{\partial \vec{w}} a_n\left(\vec{w}\right) = \sum_{n=1}^{N} \frac{\partial c_n}{\partial \vec{w}} \tag{15}$$

and therefore that the Gauss-Newton Hessian approximation becomes:

$$\mathbf{H}\left(\vec{w}\right) = \sum_{n=1}^{N} \left(\frac{\partial a_n}{\partial \vec{w}}\right)^{\mathrm{T}} \frac{\partial a_n}{\partial \vec{w}} = \sum_{n=1}^{N} \frac{1}{2c_n\left(\vec{w}\right)} \left(\frac{\partial c_n}{\partial \vec{w}}\right)^{\mathrm{T}} \frac{\partial c_n}{\partial \vec{w}} \tag{16}$$

i.e., it introduces a factor $1/2c_n(\vec{w})$ compared to how one would "normally" calculate the Gauss-Newton Hessian.

The gradient $\nabla C(\vec{w})$ is a $3M \times 1$ column vector (where M is the number of B-splines used to represent one directional component of the warp-field), whose m^{th} element is of the form:

$$\nabla C_m = \sum_{n \in B_m} \underbrace{\left[\frac{\partial c_n}{\partial \vec{s}_n}\right]}_{1\times 3} \underbrace{\left[\frac{\partial \vec{s}_n}{\partial \vec{\mathbf{J}}_n}\right]}_{3\times 9} \underbrace{\left[\frac{\partial \vec{\mathbf{J}}_n}{\partial w_m}\right]}_{9\times 1} \tag{17}$$

Where:

$$B_m = \{\text{voxels in } \mathbf{g} \text{ where B-spline } m \text{ has support}\}$$

where $n \in B_m$ means that the summation is over all voxels n for which the B-spline m has support, and where $\vec{\mathbf{J}}_n$ is a vectorised version of the Jacobian at the nth voxel and $\vec{s}_n$ is a vector with the three singular values of that Jacobian.

Similarly the Gauss-Newton Hessian is a $3M \times 3M$ matrix whose jk^{th} element is:

$$\mathbf{H}_{jk} = \sum_{n \in \{B_j \cap B_k\}} \frac{1}{2c_n(\vec{w})} \underbrace{\left(\underbrace{\left[\frac{\partial c_n}{\partial \vec{s}_n}\right]}_{1\times 3} \underbrace{\left[\frac{\partial \vec{s}_n}{\partial \vec{\mathbf{J}}_n}\right]}_{3\times 9} \underbrace{\left[\frac{\partial \vec{\mathbf{J}}_n}{\partial w_j}\right]}_{9\times 1} \right)}_{\text{scalar}} \underbrace{\left(\underbrace{\left[\frac{\partial c_n}{\partial \vec{s}_n}\right]}_{1\times 3} \underbrace{\left[\frac{\partial \vec{s}_n}{\partial \vec{\mathbf{J}}_n}\right]}_{3\times 9} \underbrace{\left[\frac{\partial \vec{\mathbf{J}}_n}{\partial w_k}\right]}_{9\times 1} \right)}_{\text{scalar}} \tag{18}$$

Where:

$B_j = \{\text{voxels in } \mathbf{g} \text{ where B-spline } j \text{ has support}\}$

$B_k = \{\text{voxels in } \mathbf{g} \text{ where B-spline } k \text{ has support}\}$

where $n \in \{B_j \cap B_k\}$ denotes summation over all voxels n for which both B-spline j and k have support. It should be noted that although $3M \times 3M$ can be a very large number, especially for high warp-resolutions/small knot-spacings, the vast majority of elements are zero because most spline combinations jk have no overlap in support.

We will demonstrate the practicalities of actually calculating these entities in the sections which follow.

2.4. Implementation

We now present how we achieved computational tractability of the SPRED penalty, its gradient and GN Hessian.

2.4.1. Parallelising Calculation of Gradient and Hessian

What may not be immediately obvious is that calculating the value of the SPRED penalty is not the costly part of this algorithm. Rather, it is the corresponding gradient and Hessian calculations which are massively computationally complex. See, for example, Equation 17, where we have multiple applications of the chain rule to vector valued functions of vectors, leading to intermediate matrices. These intermediates then need to be multiplied together, further increasing the complexity. It is easy to see that this complexity is compounded even further in Equation 18 when calculating the Hessian.

If these calculations were all performed sequentially the problem would become computationally intractable, and the suggested form of regularisation would be a mathematical nicety with no practical impact.

However we will demonstrate how the problem can be framed in terms of the application of multiple massively parallelised SIMD operations, rendering it eminently tractable.

Let us consider how one might parallelise the gradient calculation in Equation 17. Firstly we note that the part:

$$\underbrace{\underbrace{\left[\frac{\partial c_n}{\partial \vec{s}_n}\right]}_{1\times 3}\underbrace{\left[\frac{\partial \vec{s}_n}{\partial \vec{\mathbf{J}}_n}\right]}_{3\times 9}}_{1\times 9} \tag{19}$$

depends only on entities dependent on which voxel n is currently being considered. Each of these 1×9 vectors can therefore be calculated and stored independently by one GPU thread. We represent these pre-calculated vectors as an $N \times 9$ matrix (where N is the total number of voxels in image $\mathbf{g}$) which we denote $[\partial c/\partial \mathbf{J}]$ where the nth row is:

$$\left[\frac{\partial c}{\partial \mathbf{J}}\right]_{n*} = \underbrace{\left[\frac{\partial c_n}{\partial \vec{s}_n}\right]\left[\frac{\partial \vec{s}_n}{\partial \vec{\mathbf{J}}_n}\right]}_{1\times 9} \tag{20}$$

The columns of $[\partial c/\partial \mathbf{J}]$ can be thought of as *partial derivative images* and can be visualised as will be demonstrated below. Note also that in these calculations one can use the approximation given in equation 13 to avoid explicit computation of the JSVs.

The next part:

$$\underbrace{\left[\frac{\partial \vec{\mathbf{J}}_n}{\partial w_m}\right]}_{9\times 1} \tag{21}$$

is "constant" in the sense that its elements are given by the spatial derivatives of the B-spline basis-functions and can be calculated (analytically) once and for all for a given knot-spacing. If we denote a column-vectorised version of a single 3D B-spline basis-function (with dimensions are given by the knot-spacing) by $\mathbf{B}$ we can create a matrix $[\partial \mathbf{J}/\partial w]$ containing all the elements we need as:

$$\left[\frac{\partial \mathbf{J}}{\partial w}\right] = \underbrace{\left[\begin{array}{ccc} \mathbf{B}^{(x)} & \mathbf{B}^{(y)} & \mathbf{B}^{(z)} \end{array}\right]}_{N_S \times 3} \tag{22}$$

where $\mathbf{B}^{(*)}$ denotes the a spline basis-function spatially differentiated in the ith direction and N_S denotes the total size of the 3D spline (determined by the knot-spacing).

Finally each element ∇C_m of the gradient can be calculated by independent threads as:

$$\nabla C_m = \sum_{n \in B_m} \sum_{k \in L_m} \left[\frac{\partial c}{\partial \mathbf{J}}\right]_{nk} \left[\frac{\partial \mathbf{J}}{\partial w_m}\right]_{n'k'} \tag{23}$$

Where:

$$B_m = \{\text{voxels in } \mathbf{g} \text{ where B-spline } m \text{ has support}\}$$
$$L_m = \{\text{elements of } \mathbf{J}_n \text{ where B-spline } m \text{ has an effect}\}$$

where $k \in L_m$ denotes that k will select the three columns of $[\partial c/\partial \mathbf{J}]$ that contribute to the mth element of the gradient. To be concrete, if $1 \leq m \leq M$ the element pertains to the x-component of the warp-field and k will select the columns of $[\partial c/\partial \mathbf{J}]$ corresponding to the first row of $\mathbf{J}$ (see equation 4), and if $M < m \leq 2M$ it will select the columns of $[\partial c/\partial \mathbf{J}]$ corresponding to the second row etc. The $'$ superscripts on n' and k' indicate transformed versions of n and k such that n' takes into account the offset of the mth spline into the image, and k' will select the corresponding column (regardless of row) in $\mathbf{J}$.

So, in summary the calculation of the gradient can be divided into three parts:

- One which is separable over voxels (or more generally "sample points") where the calculations for one voxel can be performed and stored independently by one GPU thread.
- One which is "constant" and can be performed once and for all and stored where the results can be accessed by all threads.
- One which is separable over warp parameters (spline coefficients) where calculations for one warp parameter can be performed and stored independently by one GPU thread.

Following the same line of reasoning, we may similarly divide the Hessian calculation of Equation 18 into portions which are constant, separable over samples, and separable over warp parameters.

As it is this process which is central to how using the SPRED penalty is made tractable, we will attempt to provide an intuitive feel for what implementing the parallelisation of equation 17 actually looks like in practice by considering a simple 2D example.

2.4.2. 2D Example

We begin by defining a displacement warp field and its values in the underlying image space. The image space is a square with dimensions 21×21. The cubic B-splines are placed with their control points (knots) 5 samples apart. Each spline has a spatial support radius of 2 knots (10 samples), and therefore we place an extra ring of splines around the perimeter of the image space, such that each point in the image has an equal number of splines with support there. As a result, our x-warp and y-warp parameters are of dimension 7×7. We initialise the warp with a random set of parameters. Figure 1 shows the warp parameters, the resulting warp in image space, and the gradient of the warp in image space.

Based on the spatial gradients of the displacement fields in Figure 1, the local Jacobian matrix can be calculated at each point in space according to Equation 5.

The elements of $\mathbf{J}$ for each point in the sample space, as well as the resulting Jacobian determinant, $|\mathbf{J}|$, are shown in Figure 2, which now contains all of the information necessary to calculate the voxelwise part (Equation 20) of the calculations for the gradient (Equation 17).

Based on the elements of $\mathbf{J}$, Figure 3(a) shows the resulting contribution to the total cost from each point in the image. 3(b) to 3(e) then constitute the gradient of the cost with respect to the elements of $\mathbf{J}$ at each point in the image, i.e., the columns of $[\partial c / \partial \mathbf{J}]$ (Equation 20). Each point in the four partial derivative images can be calculated completely independently of every other point. The calculation of these images therefore constitutes the first parallelised step of our algorithm

We now move onto visualising the result of Equation 23. Note that Equation 22 is constant across the image and identical over the support of every spline. Therefore, when calculating the gradient in a parallel-per-parameter regime, the operation conforms exactly to the SIMD paradigm and may be very efficiently implemented on GPUs. In practice, we perform one such cal-

culation for each combination of **J** element gradient image (Figure 3(b) to 3(e)) and warp direction (x or y), the results of which are shown in Figure 4(b) to 4(e). Finally, we sum over these combinations for each warp direction, and the result is then the gradient of our penalty with respect to our warp parameters. This is shown in 4(f) and 4(g).

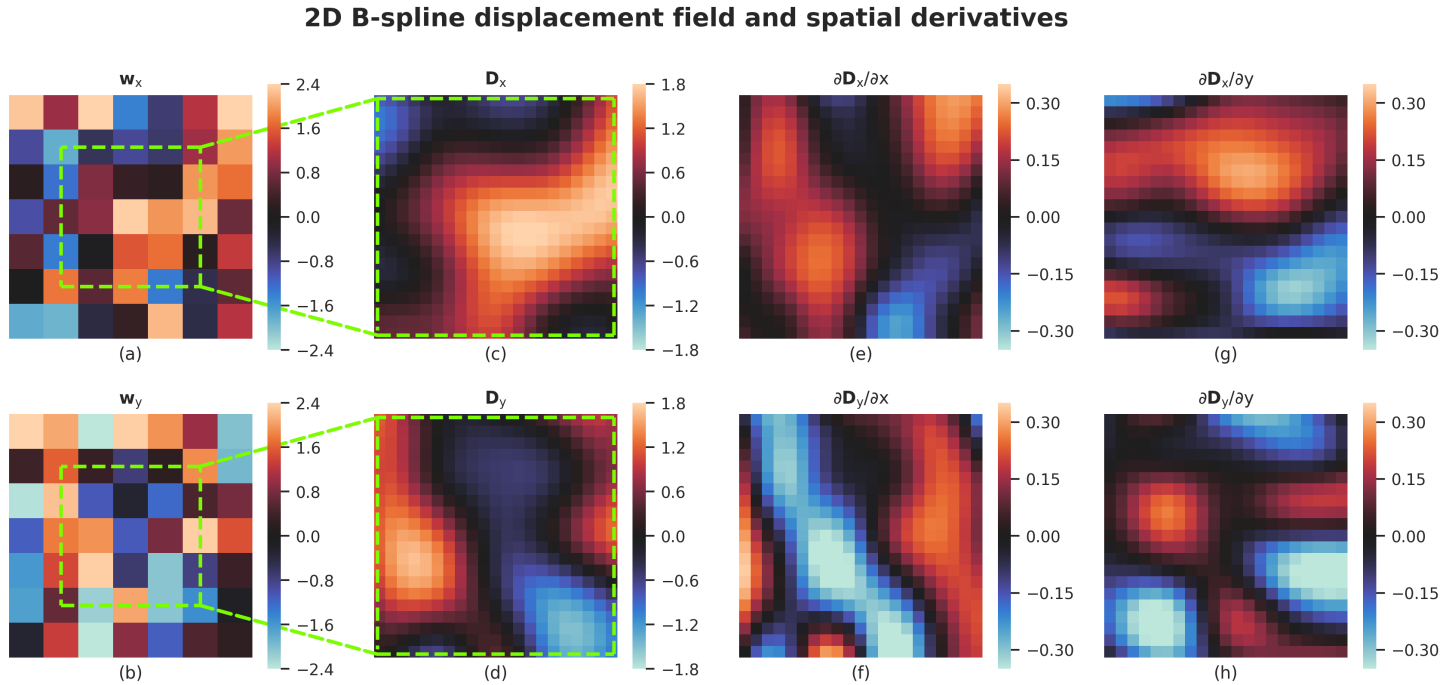

Figure 1: Example of a warp (or technically *displacement*) field for a random set of parameters (cubic B-spline coefficients). The positive x-direction is defined from left to right, and the positive y-direction from top to bottom. The x-warp parameters $\mathbf{w}_x$ (a) and y-warp parameters $\mathbf{w}_y$ (b) are arranged on a regular 7×7 grid. The resulting x-displacement field $\mathbf{D}_x$ (c) and y-displacement field $\mathbf{D}_y$ (d) corresponding to $\mathbf{w}_x$ and $\mathbf{w}_y$ are evaluated on a regular grid with knot spacing of 5×5. Note that only those parts of the warps with full support of the surrounding B-splines are shown (i.e., the area outside of the green square is considered to be *outside* of the image we are warping). This leaves a square of 21×21 samples in image space. The spatial partial derivatives of the x-warp in the x-direction (e), y-warp in the x-direction (f), x-warp in the y-direction (g) and y-warp in the y-direction (h) are presented, as these values are used in determining the local Jacobian matrices of the warp field. Cubic B-spline parametrisation ensures that both the warps and their spatial derivatives have analytical solutions and are smooth-continuous.

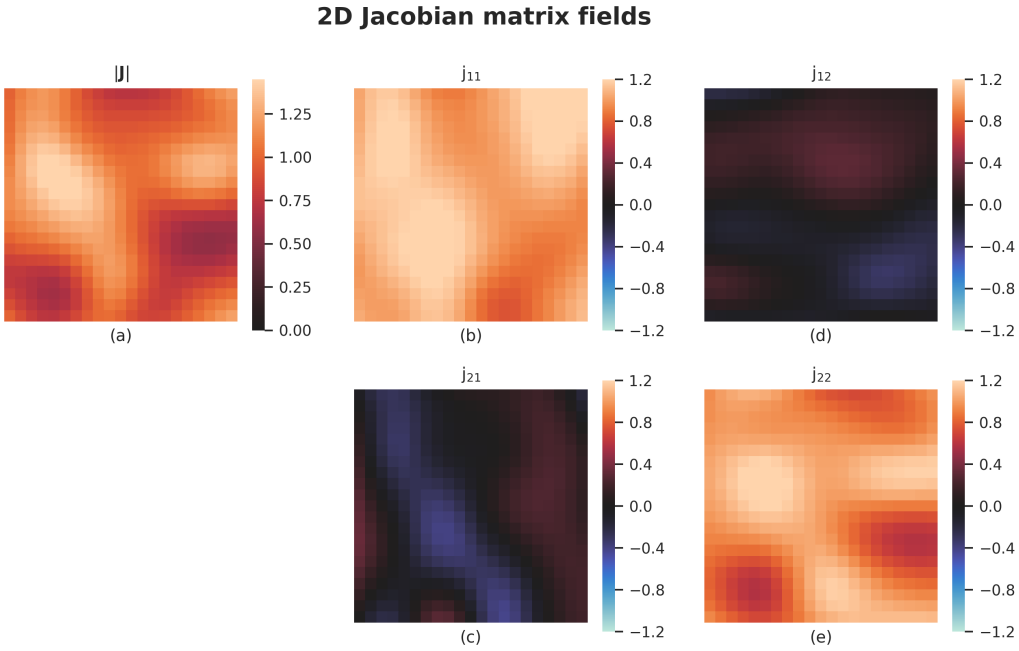

Figure 2: Elements of the local Jacobian matrix of the warp. At each position $[x, y]$ in image space a local Jacobian matrix $\mathbf{J}$ is calculated according to Equation 5 by adding the identity matrix I_2 to the 2×2 matrix composed of the elements of Figure 1 (e) to (h) at that position. The resulting 2×2 2D fields (each showing how one element of $\mathbf{J}$ varies across the image) are presented here in sub-figures (b) to (e). From these a local Jacobian determinant field (a) can be calculated. The SPRED penalty is based on the singular values of $\mathbf{J}$, and is therefore a function of fields (b) to (e).

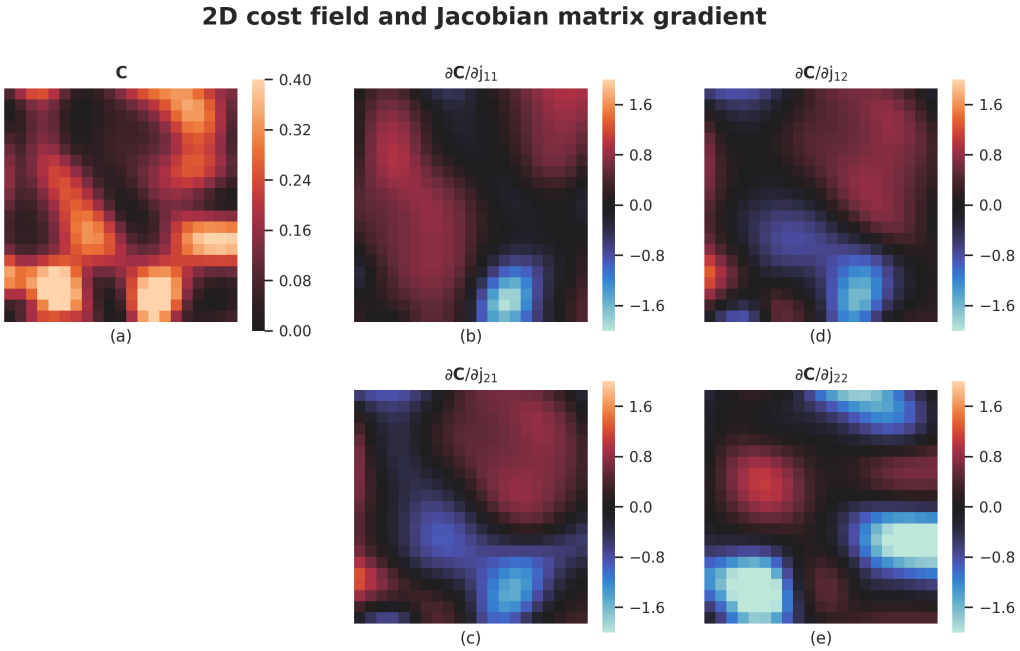

Figure 3: From the singular values of the local Jacobian matrices of Figure 2 a cost field $\mathbf{C}$ (a) can be evaluated at each position $[x, y]$ of the image. $\mathbf{C}$ then represents the contribution to the total cost from each point in image space. The partial derivatives of $\mathbf{C}$ with respect to the elements of the $\mathbf{J}$ are shown in sub-figures (b) to (e). These partial derivative fields (which together make up the overall gradient $\partial\mathbf{C}/\partial\mathbf{J}$), can be calculated independently for each position $[x, y]$ as they depend *only* on the values of $\mathbf{J}$ at $[x, y]$, making this step ideal for a parallel-per-voxel approach.

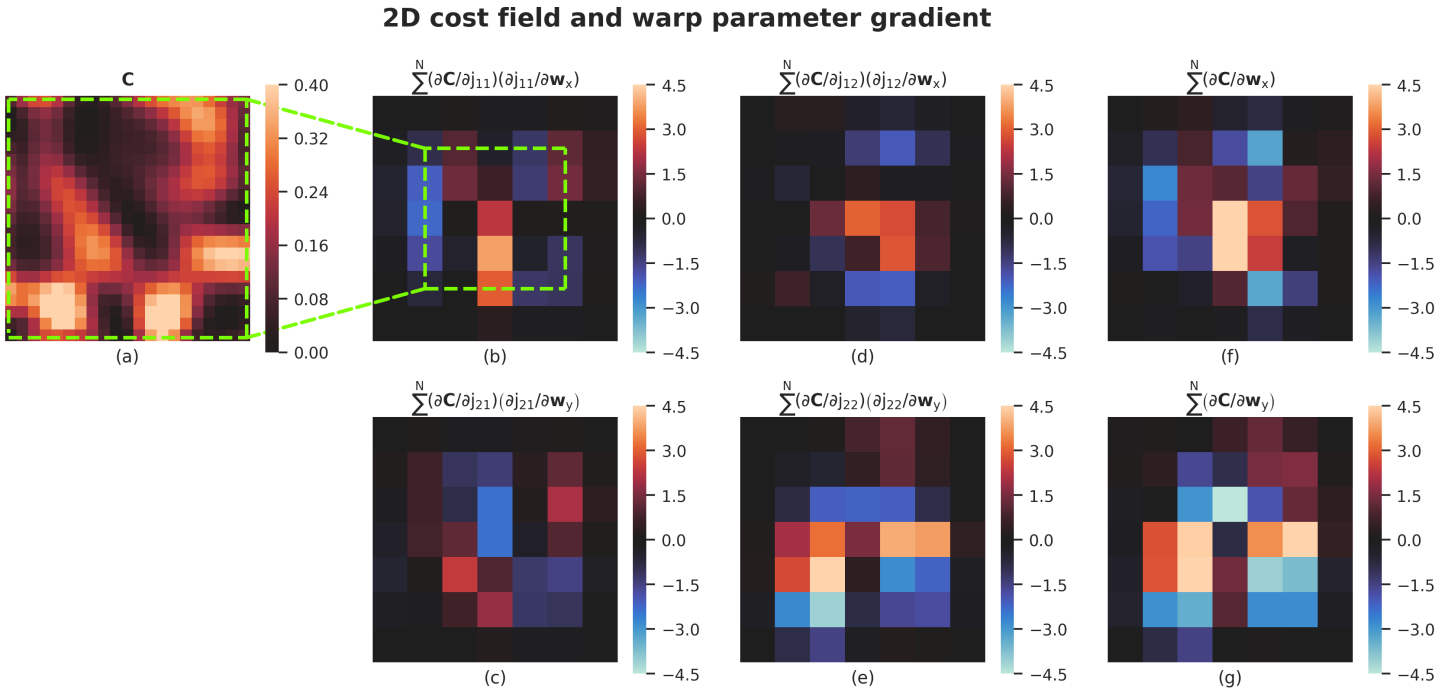

Figure 4: The final, and most computationally expensive, step is calculating the gradient of the total SPRED penalty (being the sum of $\mathbf{C}$ over all x and y) with respect to the warp parameters. Given that we have the gradient of $\mathbf{C}$ with respect to $\mathbf{J}$, knowing the gradient of the resulting partial derivative fields (i.e., Figure 3 (b) to (e)) with respect to the warp parameters would allow us to apply the chain rule to calculate this. We proceed by recognising that the piece-wise polynomial nature of B-splines means that the gradient of $\mathbf{J}$ with respect to each warp parameter is constant. Additionally, the gradient at position $[x, y]$ is exactly 0 for any B-spline which does not have support at that position. Thus, we can calculate the total gradient with respect to each warp parameter by summing the result of the chain rule across each position $[x, y]$ where that spline has support. In (b) to (e) we can see this illustrated in terms of the contributions to the overall gradient due to the effect of the warp parameters $\mathbf{w}_x$ and $\mathbf{w}_y$ on each element of the $\mathbf{J}$ fields. These partial derivatives can be calculated independently for each warp parameter, making this step ideal for a parallel-per-parameter approach. The end result of the process is then the gradient of the total penalty with respect to $\mathbf{w}_x$ (f) and $\mathbf{w}_y$ (g).

2.5. Testing

As assessing the performance of a registration tool is non-trivial, we here try to strike a balance between considering some measure of data-consistency, and some measure of anatomical plausibility. It is known that image similarity is a poor choice for measuring registration accuracy as it is just a proxy for what we are truly interested in, i.e., alignment of common anatomical structures (Rohlfing, 2012). Image-wide tissue maps are not a good option either, as they are global in nature and only really test tissue classification rather than anatomical alignment (Rohlfing, 2012). We therefore choose the overlap of segmented cortical regions as our data-consistency measure, which is an established method by which to assess the quality of registration (Klein et al., 2009).

The focus of this work is not primarily on assessing our method's ability to maximise data consistency. However, if it was unable to produce high quality overlap results then its usefulness would be severely limited irrespective of how plausible the deformation is. High registration accuracy therefore simply allows us to compare anatomical plausibility in a meaningful way.

Two main methods of assessing anatomical plausibility are employed here. One is to consider the histogram of Jacobian-determinants within the brain, which provides information about how aggressively the warp is squeezing or expanding the volume on average. The second is to observe how the Jacobian-determinants are spatially distributed, i.e., whether the deformation appears to be spatially sensible.

2.5.1. Registration Tools

In order to make a meaningful and relevant evaluation of the effect of the SPRED penalty on registration performance, four volumetric registration tools tools were included in this evaluation, namely:

- FLIRT (Jenkinson and Smith, 2001)
- FNIRT (Andersson et al., 2007)
- ANTs (Avants et al., 2008)
- MMORF

Whilst there are myriad other tools which may also have been included in this comparison (e.g., SPM Unified Segmentation (Ashburner and Friston, 2005), DARTEL (Ashburner, 2007), elastix (Klein et al., 2010), MIRTK

(Rueckert et al., 1999; Schnabel et al., 2001), Diffeomorphic Demons (Vercauteren et al., 2009)) the chosen methods represent a broad enough overview of the types of tools available for the purposes of characterising the performance of MMORF.

FLIRT (the only linear tool tested) was included as a reference point for assessing the baseline image similarity before applying any of the nonlinear registration tools.

FNIRT was included as the reference elastic framework registration tool. It shares a similar transformation model and similarity metric with MMORF (3D cubic B-spline and mean-squares), but employs the commonly used Bending Energy (BE) regularisation penalty as default. Additionally, FNIRT attempts to preserve topology by projecting the final deformation field onto the closest diffeomorphic B-spline representation, thereby overcoming some of the perceived limitations of elastic frameworks presented in Section 1.

ANTs (specifically antsRegistration which forms part of the ITK software library (Avants et al., 2014)), was included as a widely used plastic framework registration tool. ANTs has previously been the best performing tool in similar comparisons, and therefore is the obvious choice in this instance (Klein et al., 2009; Ou et al., 2014). We have included two variants of ANTs, which differ only in the choice of similarity metric. The first variant uses MSQ, and was included in order to reduce the differences in implementation between the ANTs and the two elastic tools. The second variant uses Cross Correlation (CC), as it is ANTs' recommended intra-modal metric and is the method which has previously been shown to produce the best overlap scores (Avants et al., 2011). In both cases the greedy SyN transformation model was used.

2.5.2. Registration Parameter Selection

An important aspect when comparing registration algorithms is ensuring that the user-selectable parameters are as optimal as possible (Klein et al., 2009).

FNIRT was run with an optimised strategy proposed by the tool's author. This consisted of a multi-resolution, multi-smoothing level registration, to a final knot spacing of 1 mm isotropic, and is explained in detail in Andersson et al. (2019).

ANTs-CC was run using the antsRegistrationSyn.sh script, but with the gradient step and smoothing values changed to match those supplied by the tool's author for use in previously published comparative studies (Klein et al.,

2009). ANTs-MSQ was run with hand tuned parameters. Both versions of ANTs employed a multi-resolution, multi-smoothing level approach, to a final resolution of 1 mm isotropic.

MMORF was run using parameters which were shown to perform well during the development of the tool. Again, a multi-resolution, multi-smoothing level approach was used, but with a final knot spacing of 1.25 mm.

Note that whilst significant effort has been made to ensure that each tool performs as well as possible, the overarching goal was not to definitively classify which tool performed best, but rather to investigate how the SPRED penalty affects the nature of the deformations when compared to similar performing tools.

2.5.3. Test Dataset

Registration algorithm testing tends to focus on fairly artificial data, however these data do serve to highlight behavioural aspects of the algorithms, and usually have some form of *ground truth*. Here we have chosen to use the Non-rigid Image Registration Evaluation Project (NIREP) dataset (Christensen et al., 2006) in testing the registration methods. This dataset consists of the T_1-weighted scans of 16 healthy subjects (8 male and 8 female). Each subject's scan has been brain extracted, bias corrected, and 32 cortical regions (16 left hemisphere, 16 right) have been expertly hand-segmented. The segmented regions provide a *ground truth* for anatomical correspondence of a finer granularity than simple tissue maps.

2.5.4. Evaluation Strategy

The registration tools were were evaluated using pairwise registrations of each combination of subjects (240 warps in total). Each of the 32 segmented regions were then transformed through these warps and various overlap metrics calculated, namely: Jaccard Coefficient, Dice Coefficient, Specificity, Sensitivity. The distributions of these overlaps in each region were then compared between registration methods. Additionally, the within-brain histogram of Jacobian determinants was calculated for selected volumes in order to gain some insight into how aggressive each method is in terms of volume changes. Finally, a visual inspection of the spatial maps of the Jacobian determinants was calculated for these volumes, in order to assess how anatomically believable these deformations were.

3. Results

3.1. Overlap Scores

Combined left/right hemisphere overlap scores are shown in Figure 5. All overlap measures showed similar trends, and therefore we focus only on the Jaccard Coefficient as our measure of choice. FLIRT, as expected, achieves the lowest levels of overlap, with mean overlaps varying depending on the region being considered. All of the nonlinear methods, again as expected, improve the degree of overlap considerably. ANTs-MSQ has the lowest average overlap of the nonlinear methods. FNIRT has the second lowest overlap scores, although in certain areas it is outperformed by ANTs-MSQ. Note that the FNIRT results are significantly better than previously reported attempts (e.g. Ou et al. (2014) who used the parameters from Klein et al. (2009) which were suboptimal when applied to the NIREP dataset), but are in line with those from the tool's author (Andersson et al., 2019). ANTs-CC and MMORF perform the best and equally well on average, but one or the other performs better in individual areas. This suggests that these two methods allow more freedom of deformation which better models complex differences in individual anatomy. These observations are supported when one considers the overall distributions of overlap scores in Figure 6 which display the same trends.

It is interesting that the two versions of ANTs produce such different results, given that they differ only in their choice of similarity metric. The first conclusion one might draw is that ANTs-CC is able to capture some element of image similarity that ANTs-MSQ cannot. However, given that MMORF is able to match ANTs-CC in overlap performance whist using an MSQ similarity metric suggests that this is not the case. An alternative explanation is that the relative weightings of the similarity metric and the level of regularisation (or more specifically their gradients) differs between these methods, resulting in a deformation which is less regularised when using CC.

The joint distribution of overlap scores between MMORF and ANTs-CC for all regions and warps shown in Figure 7 confirms that the two methods are on average very comparable. If judging the quality of registration by this metric alone then there would be no meaningful difference between them.

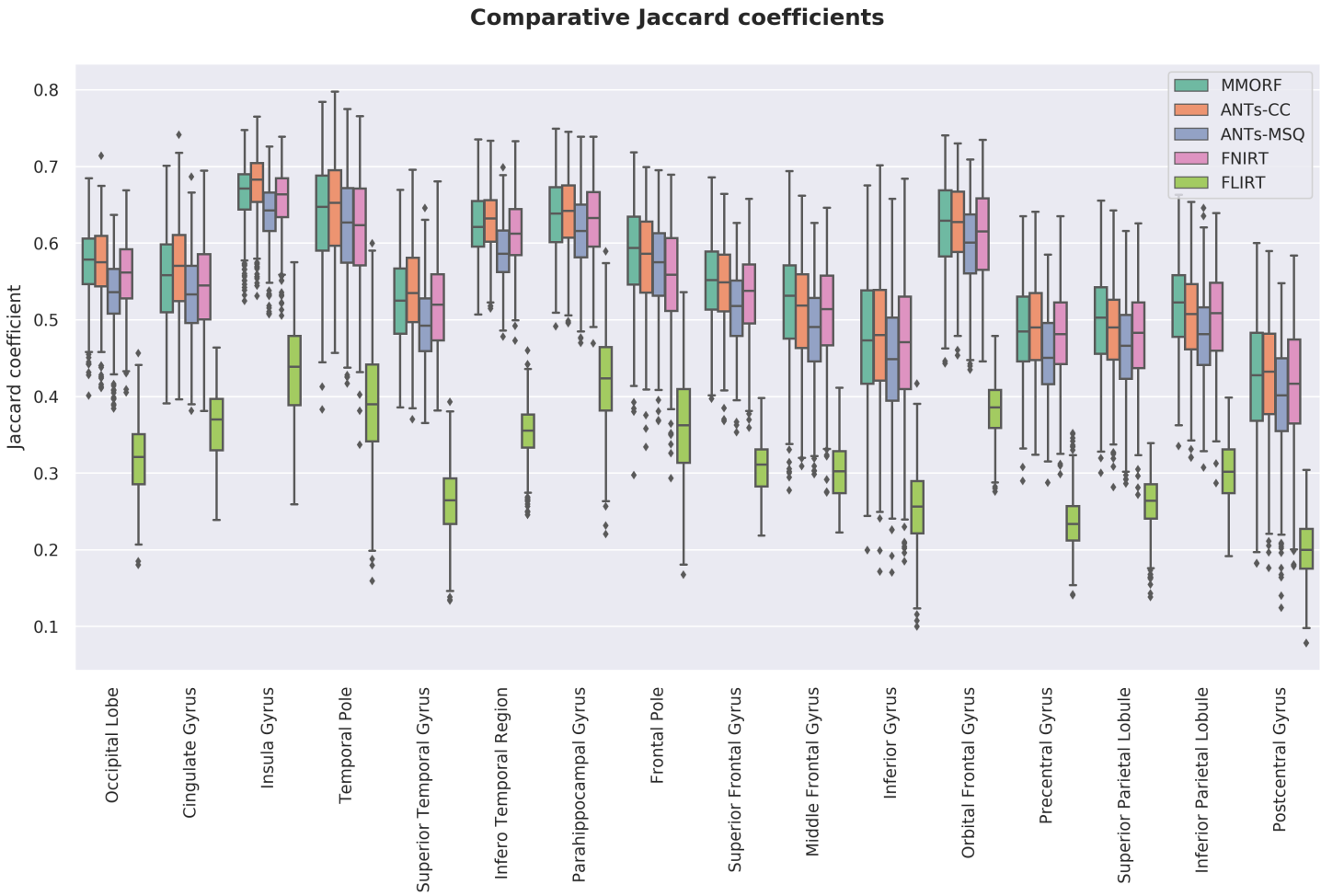

Figure 5: Jaccard coefficients of 16 regions (32 original cortical segmentations combined left/right) for each of the 4 nonlinear and 1 linear registration methods. The FLIRT result provides a baseline for evaluating the improvement in overlap between linear and nonlinear methods. Overall ANTs-CC and MMORF perform similarly. In some areas ANTs-CC produces better results (e.g., cingulate gyrus), and in some area MMORF performs better (e.g., inferior parietal lobule). In all cases, both methods outperformed ANTs-MSQ and, to a lesser extent, FNIRT.

3.2. Jacobian Determinant Distributions

Figure 9 shows the Jacobian determinant distribution of the warps for a randomly selected image pair. We show both the non-log and log distributions. The former is informative for showing the behaviour of the tools for determinants approaching zero, whilst the latter is more informative for showing how compression and expansion differ.

In non-log form, FNIRT has the most distinctive shape, with a linear tailing off of values approaching zero, as opposed to a sharp drop from the other methods. This is most likely due to FNIRT projecting its final warp onto the nearest B-Spline field which has no negative Jacobian determinants. In log-form, all methods have a similar shape, being largely symmetric about 0 with higher magnitudes becoming increasingly unlikely. FNIRT has the widest distribution of log-Jacobian determinants, followed by ANTs-CC. ANTs-MSQ and MMORF produce very similar distributions, which are both narrower than the other methods.

From the log distribution, the range of the 5th to 95th percentile for each method can be calculated. This was done for every warp, and the results are summarised in Figure 10. We have included only MMORF, ANTs-CC and ANTs-MSQ as this covers methods with similar overlap performance (MMORF and ANTs-CC) and similar single subject Jacobian determinant distributions (MMORF and ANTs-MSQ). ANTs-CC shows significantly higher average ranges compared to MMORF and ANTs-MSQ. MMORF and ANTs-MSQ have very similar average performance, but with ANTs-MSQ showing higher variability.

So, on this metric ANTs-MSQ outperformed ANTs-CC. Note that this is the opposite to what we observed for the overlap scores, where ANTs-CC had superior, and more similar to MMORF, results.

Figure 11 presents a direct comparison between percentile ranges for each image pair between ANTs-CC and MMORF. In all but 4 cases the percentile range is smaller for MMORF.

3.3. Jacobian Determinant Spatial Maps

Figure 12 shows a map of the spatial distribution of log-Jacobian determinants for the same randomly selected pair of subjects as Figure 9. Here we present only the maps of the best performing methods in terms of overlap scores (MMORF and ANTs-CC). There are clear visual differences between the results of two methods. In areas of variable T_1-weighted signal intensity, such as the cortex, both methods produce similar looking maps with a varying

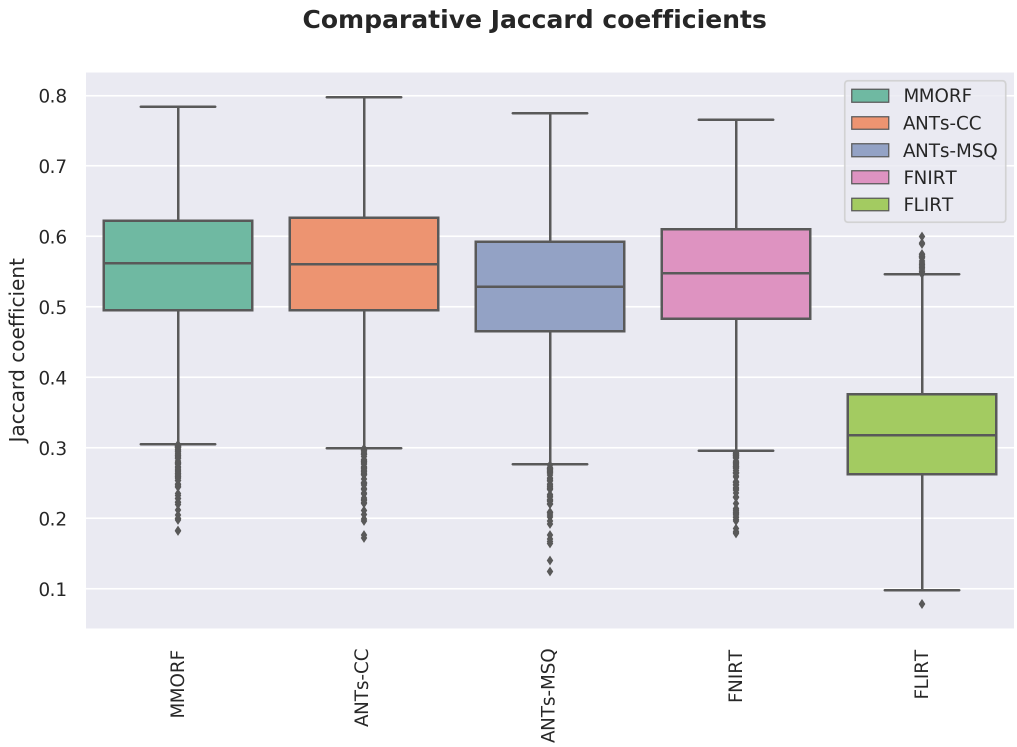

Figure 6: Distribution of Jaccard coefficients combined across all regions and subject pairs for each of the 4 nonlinear and 1 linear registration methods. This confirms the trends observed in Figure 5, with MMORF and ANTs-CC performing best, followed by FNIRT and then ANTs-MSQ. The MMORF and ANTs-CC distributions are remarkably similar, despite the slight region-to-region differences in performance.

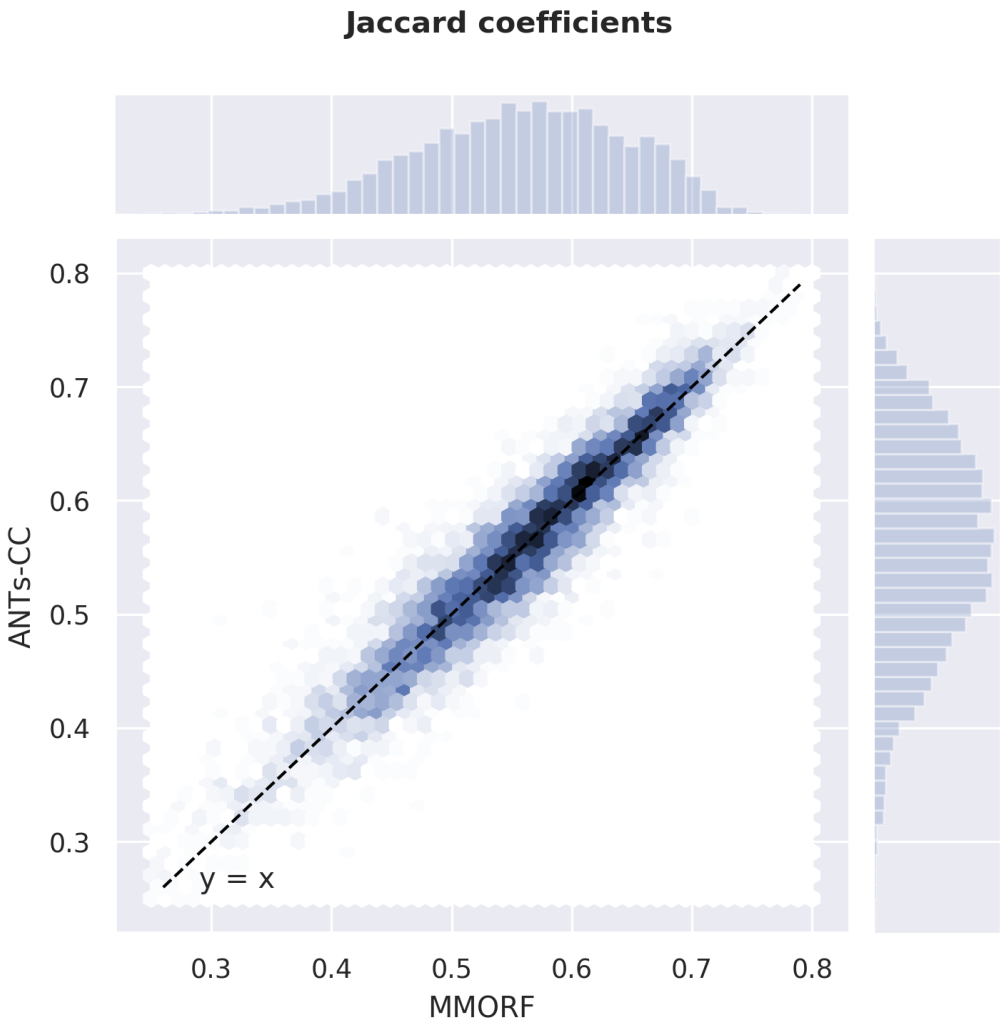

Figure 7: Joint histogram of Jaccard coefficients across all regions and warps for MMORF and ANTs-CC. Note the tight, linear relationship, supporting the observation from Figure 5 that MMORF and ANTs-CC have very comparable overall performance across the NIREP dataset.

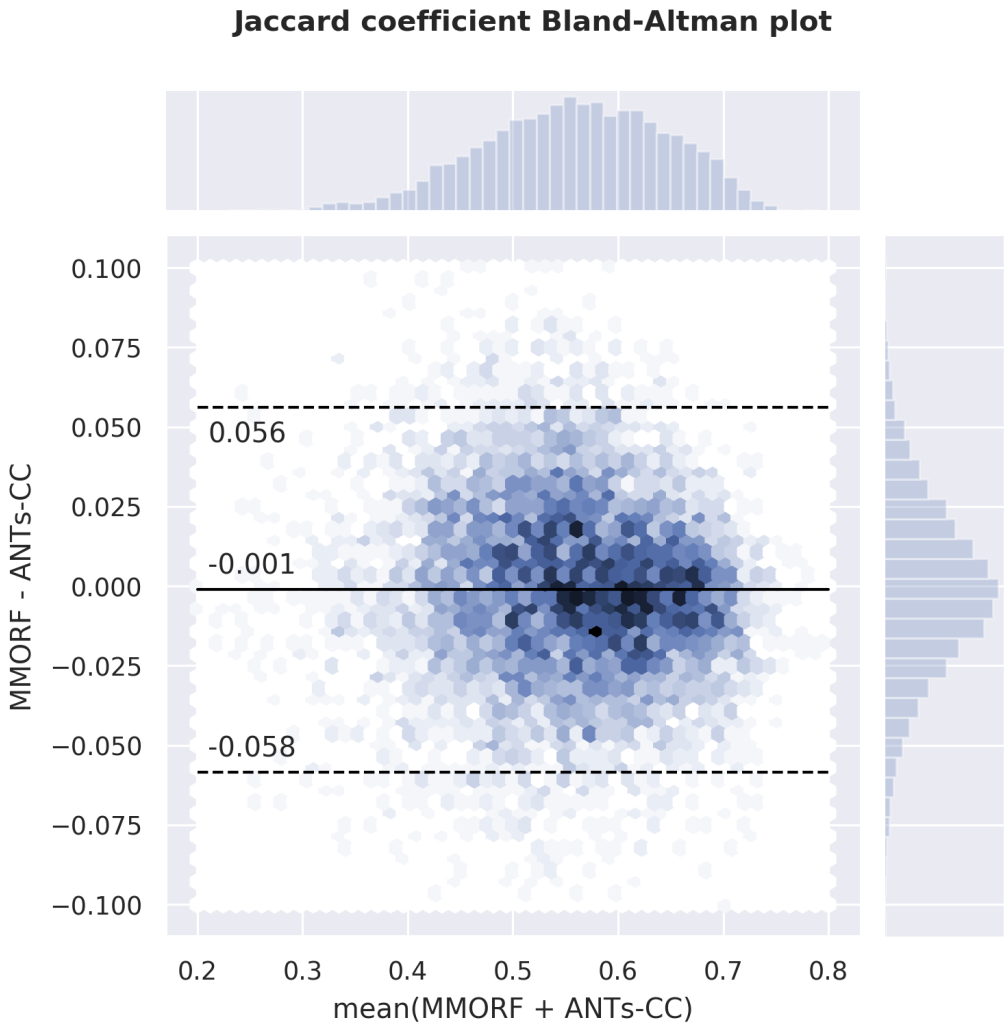

Figure 8: Bland-Altman plot of Jaccard coefficients across all regions and warps for MMORF and ANTs-CC. The mean difference between the methods (-0.001) is negligible, indicating no systematic difference between the two methods. The +-1.96 standard deviation range of the difference is reasonably tight (-0.058 to 0.056), indicating that when the methods do differ, that those differences tend to be small. Overall we may then conclude that these two methods can be considered interchangeable in terms of registration accuracy as measured by Jaccard coefficients.

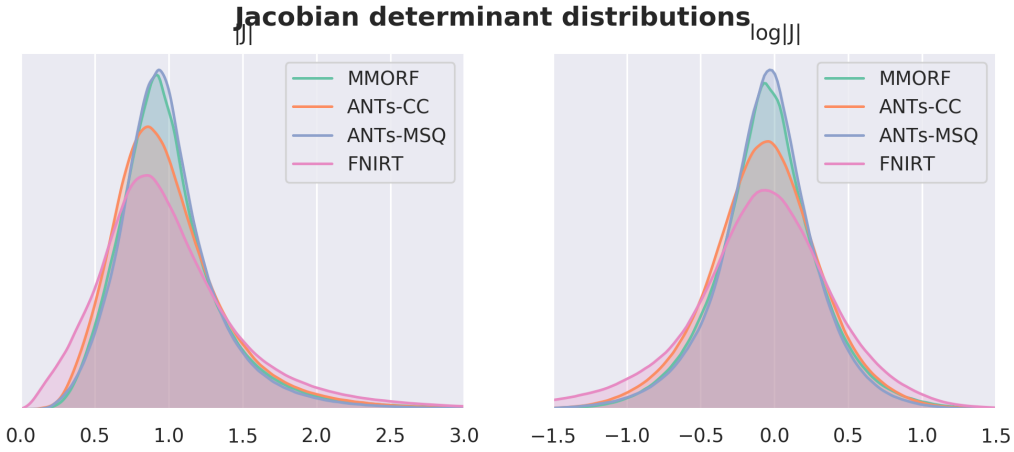

Figure 9: Jacobian determinant and log-Jacobian determinant distribution of a randomly selected registration pair from the NIREP dataset. Only values within the brain itself have been considered. As desired, all four methods produce Jacobian determinants with few to no negative values. There are distinct differences in appearance though. FNIRT is the most different, with a visibly heavier negative log-Jacobian tail. This is an effect of projecting the Bending-Energy regularised registration onto a field without negative Jacobian determinants. ANTs-CC displays the next widest log-Lacobian distribution, but with more evenly weighted tails. ANTs-MSQ and MMORF have distributions with very similar widths, indicating comparable levels of aggressiveness in terms of volumetric changes.

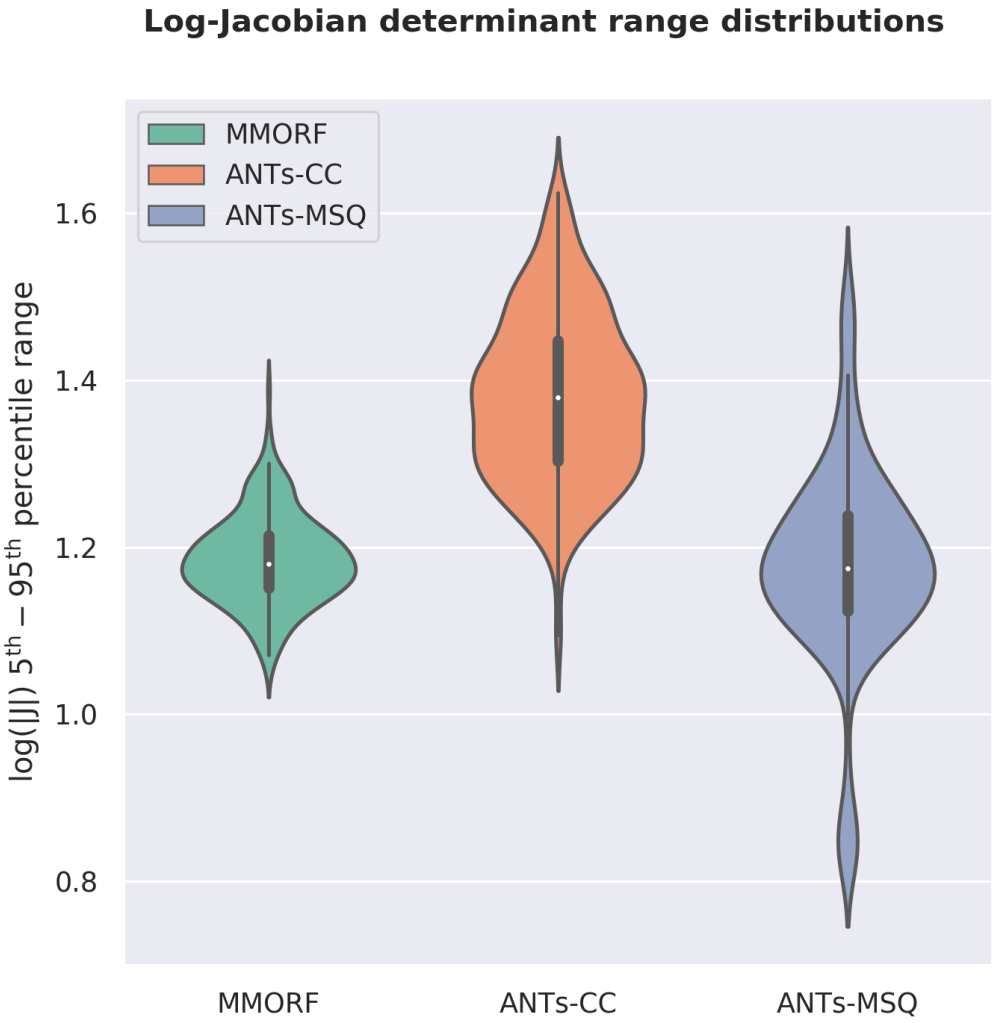

Figure 10: The 5th to 95th percentile is used as a summary measure for the log-Jacobian determinant distributions in Figure 9. The distribution of this percentile range over all 240 registrations is shown for three registration methods. ANTs-CC displays a significantly higher range on average when compared to both MMORF and ANTs-MSQ. ANTs-MSQ displays a similar average range to MMORF, but with more variability. MMORF therefore has the same or better average range, with less variability than both ANTs methods.

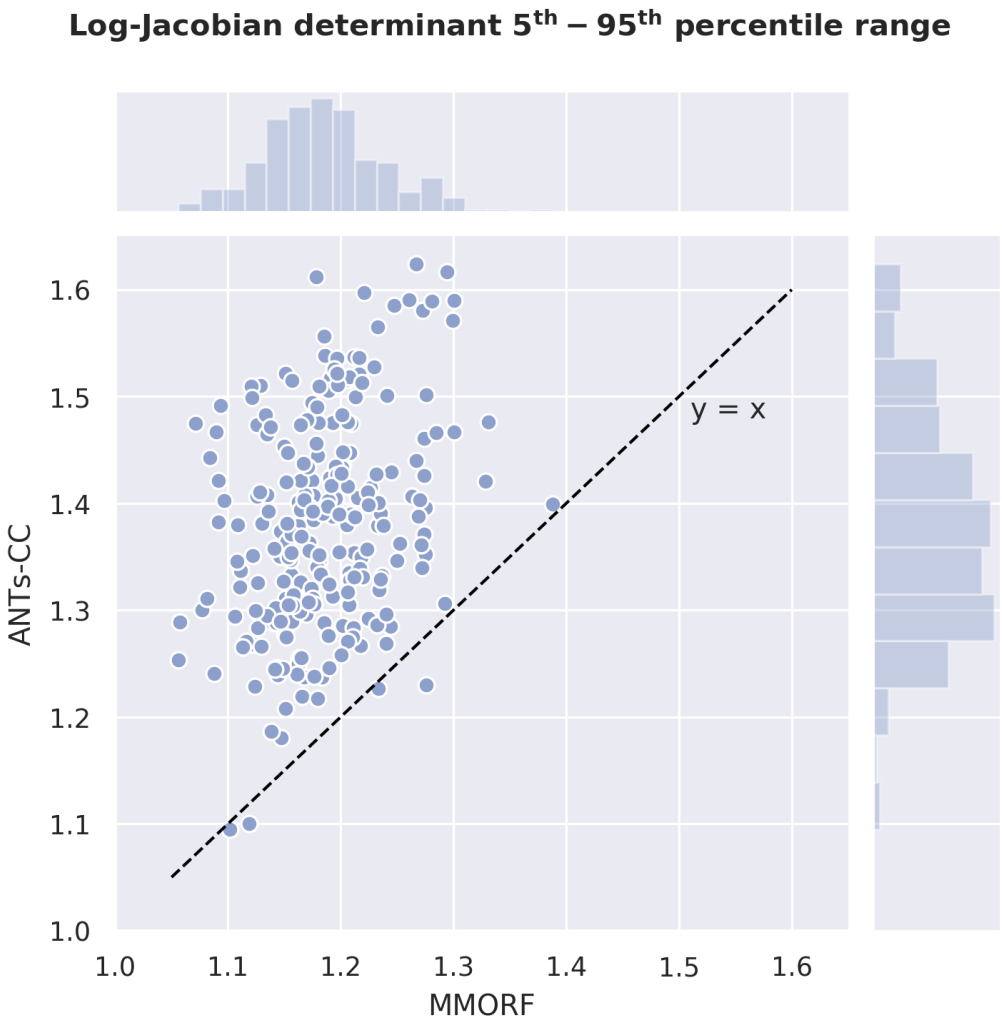

Figure 11: Scatter plot of MMORF and ANTs-CC log-Jacobian determinant 5th to 95th percentile ranges. The key feature here is that ANTs-CC produces systematically wider ranges compared to MMORF, with only 4 of the 240 registrations favouring ANTs-CC. This is despite ANTs-CC and MMORF producing comparable Jaccard coefficients overall, as shown in Figure 7.

amounts of expansion and compression (positive and negative log-Jacobian determinants). Where they differ in appearance is predominantly within areas displaying a relatively flat T_1-weighted signal profile, such as in white matter. Here MMORF produces smoothly changing values with magnitudes close to 0. ANTs-CC by contrast displays highly oscillatory behaviour, with a higher proportion of values deviating away from 0.

As a T_1-weighted volume contains little information within the white matter, there is little reason to believe that the deformation should be anything other than smooth within these regions. Clearly though this has not detrimentally affected the performance of ANTs-CC in terms of achieving high cortical overlap scores. However, if this warp was used to transform a modality rich in white-matter information (such as diffusion tensor imaging) the results could potentially be very deleterious. By contrast, the MMORF result is more balanced and anatomically plausible. In areas where there is a large amount of information to drive the deformation (tissue boundaries) the warps are correspondingly more variable, whereas areas with minimal information are left relatively unchanged

4. Discussion

4.1. Relationship to Earlier Work

The main inspiration for the current paper is the work of Ashburner et al. (Ashburner and Friston (1999) and Ashburner et al. (2000)) where they suggest the particular form of regularisation function that we have used. One important difference to that work is our use of splines to represent the deformation fields that allows us to use analytical derivatives and facilitates a more sophisticated optimisation strategy.

Although most published work has focused on "composition of warps" (*e.g.* Miller et al. (2003) and Beg et al. (2005)) as a means of ensuring diffeomorphic mappings, there is other work that has pursued a similar strategy to ours. Notably Burger et al. (2013) use a regularisation function derived from models for hyperelastic materials. These are models which describe the stress-strain relationships of materials, for example rubber, that are not well described by linear elastic models. Ruthotto et al. (2012) have also applied this hyperelastic regularisation to correcting susceptibility artefacts in diffusion-weighted MRI due to its ability to enforce diffeomorphism and ensure the warps used to undistort images remain meaningful. Similar hyperelastic models have additionally been used to model cortical growth in

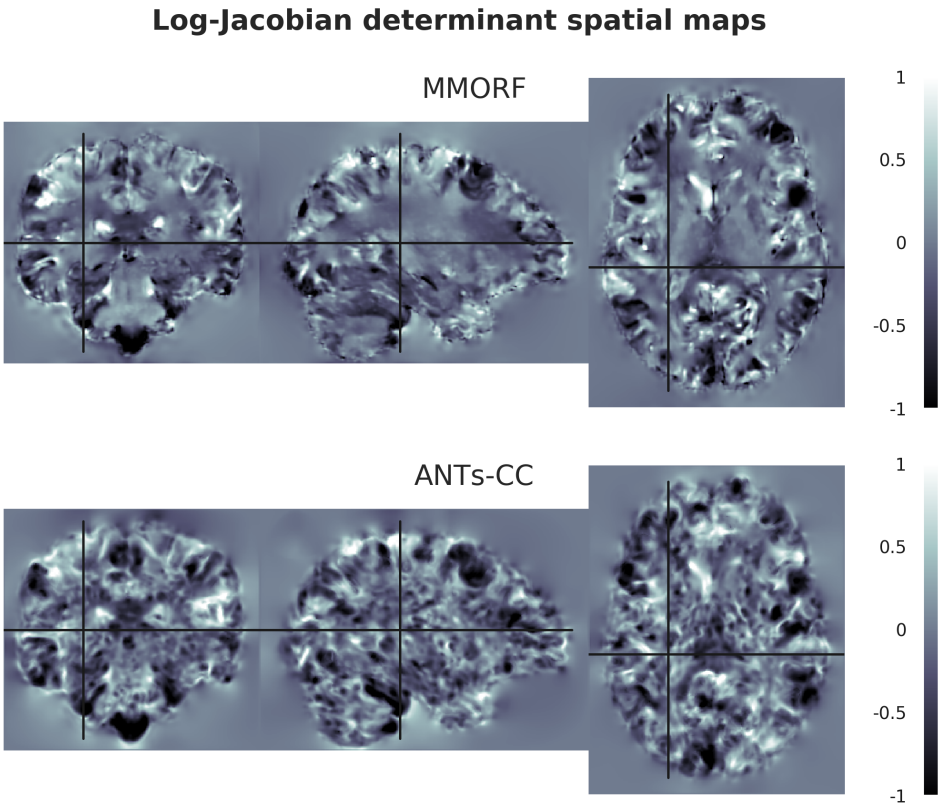

Figure 12: Spatial distribution of log-Jacobian determinants for MMORF and ANTs-CC. Note the distinct visual differences within the white matter. The SPRED penalty has resulted in smooth changes within this region, whilst ANTs-CC has produced high frequency variations. As a T_1 image contains little information within the white matter, the anatomical plausibility of those rapid variations is not clear. By maintaining smoothness within the white matter, MMORF produces a result which is more anatomically believable given the information present within the images being registered. Importantly, the irregularity introduced by ANTs-CC is not necessary for achieving high overlap scores, as Figures 5 and 7 show. Finally, whilst this may not have any deleterious effect on overlap scores of grey matter regions, the effect of transforming a modality rich in white matter information through such a warp could be significant.

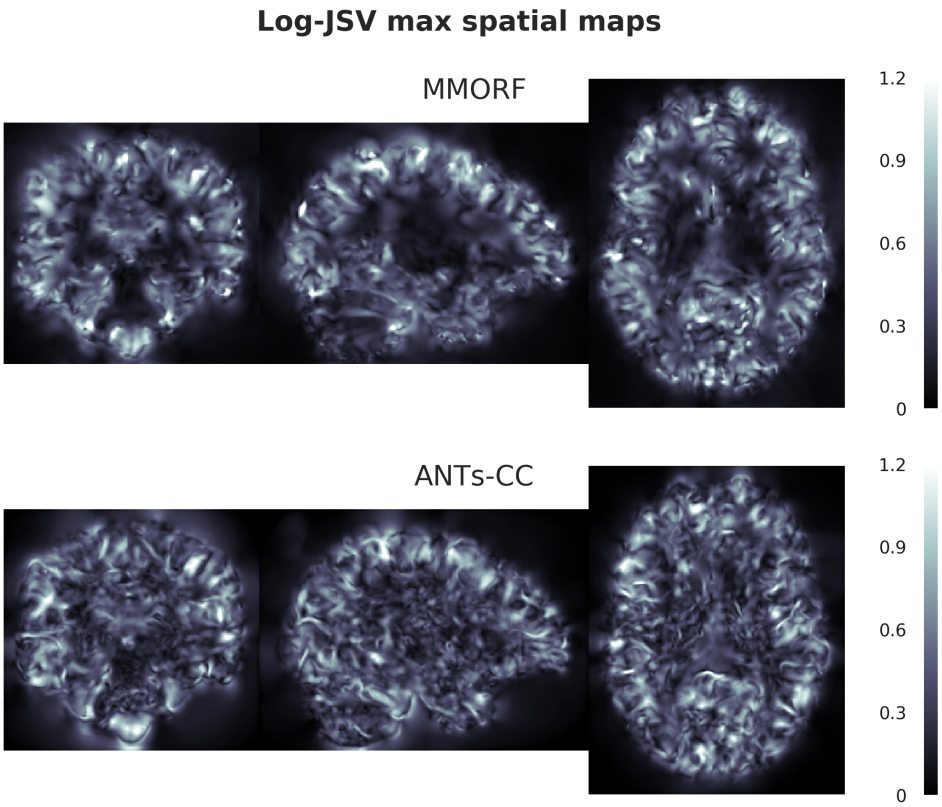

Figure 13: Spatial distribution showing the maximum of the absolute value (of the three) log-Jacobian singular values for MMORF and ANTs-CC. It represents the maximum linear relative change along any direction of the deformed voxels. Note that the appearance is very similar to that of figure 12, showing how the SPRED penalty is able to regularise linear as well as volumetric changes.

the developing brain (Knutsen et al., 2010) and for regularisation of surface based cortical registration (Robinson et al., 2018).

It should be noted that neither the present work, nor that in Ashburner et al. (2000), is based on hyperelastic models. Nor do we believe that hyperelastic models are necessarily meaningful descriptive generative models for explaining anatomical differences between subjects. Any such model (see for example Van Essen (1997)) is unlikely to be simple enough to be described by a few material constants. The reasoning behind our regularisation function is purely empirical, based on fulfilling our criteria for a useful function and on proving to produce smooth and plausible warps in parts of images with little or no anatomical information.

4.2. Registration Accuracy

As we have previously stated, registration accuracy is not the primary focus of this work. Instead it stands as a reference point for our discussion of anatomical plausibility, in that two methods which are comparable in terms of registration accuracy can be differentiated based on anatomical plausibility, and vice versa.

Based on the overlap scores in Section 3.1 we see that MMORF and ANTs-CC are clearly the best performing tools in terms of registration accuracy. Additionally there is very little to differentiate between their performance as a whole. We note that the results for ANTs-CC are slightly better than those which have been previously reported (Ou et al., 2014), and may therefore be considered as a realistic measure of the optimal performance of ANTs-CC on this dataset. From this we conclude that the small and large deformation frameworks are equally capable of achieving high registration accuracies.

4.3. Anatomical Plausibility

Based on the summary 5th to 95th percentile log-Jacobian determinant range metric, MMORF is significantly more anatomically plausible than ANTs-CC. Figure 11 supports this argument, by showing that this is true systematically. In other words, for a given registration accuracy one is almost guaranteed to have lower volumetric changes when using MMORF over ANTs-CC. Another way of framing this result is to say that larger log-Jacobian determinants are not a necessary penalty for achieving high registration accuracy.

Furthermore, MMORF produces a much tighter distribution of percentile ranges than either ANTs-CC or ANTs-MSQ. This suggests that by using the SPRED penalty we may control not only the mean percentile range, but also the allowed variance. Therefore, even though MMORF and ANTs-MSQ would on average appear to be similarly anatomical plausible, there is more chance of any given registration producing undesirably wide ranges when using the latter.

Let us now suppose the total amount of volumetric change is not a concern for a particular application, e.g., we believe we can correct for it given that we have a spatial map of the effect. In this instance we again have no reason to choose between MMORF and ANTs-CC. However, should we wish to use the structural registration to transform functional or diffusion data (i.e., data which is inhomogeneous within grey or white matter areas) then there are clear benefits to using MMORF. The high levels of spatial variability visible in the ANTs-CC warps within areas of structural homogeneity means that data in these regions would be transformed in a highly implausible manner. Consider what this might do to diffusion MRI data, where the warp would both randomly shrink and enlarge portions of the data, as well as randomly reorienting it. It is not a given that this would even be picked up on in this case, if the notion of anatomical plausibility is not extended beyond simple summary metrics such as maintaining diffeomorphism.

Why this behaviour is observed in ANTs-CC is an interesting question in and of itself. At first this might be thought to be a case of insufficient regularisation, however the deformations generated by ANTs-MSQ did not display this same behaviour. Therefore we must posit that this is due to the somewhat *scale-free* nature of the CC metric. In other words, the gradient of CC does not depend on the amount of signal contrast present, only on the local correlation of that signal. What this means in practice is that regularisation on the level of the gradient cannot alter this behaviour, rather regularisation of the metric itself would be required. How to achieve this is beyond the scope of this work, however it highlights the importance of taking a considered approach to evaluating the believability of a deformation.

4.4. Limitations of the current work

The aim of the present paper is to introduce the SPRED penalty and to show that it can be used to yield large deformation, diffeomorphic and anatomically plausible warps. It is not yet a finished registration package that deals with differences in contrast, receive bias-field or B1 inhomogeneities.

These will be the subject of future work. Neither have we applied it to data where *very* large deformations are needed, such as when registering images of severely atrophied brains or inter-species registration.

4.5. Performance Summary

The SPRED penalty allows MMORF to overcome the oft-quoted limitations of small deformation frameworks and achieve levels of registration accuracy comparable to state-of-the-art large deformation tools such as ANTs-CC. MMORF achieves this whilst at the same time producing warps which are systematically less aggressive in terms of log-Jacobian determinant ranges. Finally, given the information present in a structural scan, the spatial distribution of the log-Jacobian determinants MMORF produces appear more plausible compared to ANTs-CC, which produces variations that are not obviously necessary.

Overall then we conclude that SPRED as implemented within MMORF produces warps which can be used with a high level of confidence. MMORF's registration accuracy is on par with state-of-the-art volumetric methods, and therefore no sacrifice need be made in terms of maximising data consistency. Above all though, the improvements in anatomical plausibility should make MMORF a clear choice when using structural registration to transform functionally meaningful data.

5. Conclusion and Outlook

We have demonstrated that our SPRED penalty is capable of matching the registration accuracy (as measured using overlap scores of manually segmented cortical regions) of some of the most well established large-deformation (ANTs-CC) and small-deformation (FNIRT) registration frameworks available today. Additionally, we find that the results of using the SPRED penalty are consistently more anatomically plausible both in terms of summary metrics (5th to 95th percentile log-Jacobian determinant ranges), and in terms of sensible spatial deformations. We leverage GPU parallelisation in order to make optimisation of the penalty computationally tractable, allowing this method to be of practical benefit as part of a usable registration framework.

We have shown that using a regularisation function such as this can overcome the shortcomings of elastic small deformation frameworks, yielding the best of both worlds: large deformation, diffeomorphic warps with minimal

distortions. Future work in this area will focus on incorporating additional modalities into the MMORF framework, as well as investigating the effect of spatially varying the weighting of the SPRED penalty.

Acknowledgements

The Wellcome Centre for Integrative Neuroimaging is supported by core funding from the Wellcome Trust [grant number 203139/Z/16/Z]. The Wellcome Centre for Human Neuroimaging is supported by core funding from the Wellcome Trust [grant number 203147/Z/16/Z].

This work was supported by funding from the Engineering and Physical Sciences Research Council (EPSRC) and Medical Research Council (MRC) [grant number EP/L016052/1]; and the NVIDIA Corporation GPU grant program.

FJL is supported by The Oppenheimer Memorial Trust, St Catherine's College Oxford and The Rotary Foundation.

JLRA is supported by a Wellcome Trust Strategic Award [grant number 098369/Z/12/Z]; and the NIH Human Connectome Project [grant numbers 1U01MH109589-01 and 1U01AG05256401].

Conflicts of Interest

We have no conflicts of interest to declare.

Appendix A. Penalty Approximation

In this section we provide the derivation of the following approximation:

$$\log^2 x \approx x + \frac{1}{x} - 2 \tag{A.1}$$

Let:

$$y\left(x\right) = x + \frac{1}{x} - 2 \tag{A.2}$$

And:

$$z\left(x\right) = \log\left(x\right) \tag{A.3}$$

$$\therefore x\left(z\right) = e^z \tag{A.4}$$

Then:

$$y(z) = e^{z} + e^{-z} - 2 \tag{A.5}$$

Now, taking the Taylor expansion of $y(z)$ about $z = 0$:

$$\begin{aligned} y(z)\,|_{z=0} &\approx y(0) + \frac{y'(0)}{1!}(z-0) + \frac{y''(0)}{2!}(z-0)^2 \\ &= 0 + 0 + z^2 \\ &= z^2 \end{aligned} \tag{A.6}$$

And finally, using Equation A.3:

$$\therefore y(x) \approx \log^2 x \tag{A.7}$$

References

Amit, Y., Grenander, U., Piccioni, M., jun 1991. Structural Image Restoration through Deformable Templates. Journal of the American Statistical Association 86 (414), 376–387.

Andersson, J. L. R., Jenkinson, M., Smith, S., 2007. Non-linear registration, aka spatial normalisation. FMRIB Technial Report TR07JA2. Tech. Rep. June.

Andersson, J. L. R., Smith, S. M., Jenkinson, M., jan 2019. High resolution nonlinear registration with simultaneous modelling of intensities. bioRxiv, 646802.
URL `http://biorxiv.org/content/early/2019/05/22/646802.abstract`

Ashburner, J., 2007. A fast diffeomorphic image registration algorithm. NeuroImage 38 (1), 95–113.

Ashburner, J., Andersson, J. L., Friston, K. J., 1999. High-dimensional image registration using symmetric priors. NeuroImage 9 (6 Pt 1), 619–28.

Ashburner, J., Andersson, J. L., Friston, K. J., 2000. Image registration using a symmetric prior–in three dimensions. Human brain mapping 9 (4), 212–225.

Ashburner, J., Friston, K. J., 1999. Nonlinear spatial normalization using basis functions. Human Brain Mapping 7 (4), 254–266.

Ashburner, J., Friston, K. J., 2005. Unified segmentation. NeuroImage 26 (3), 839–851.

Ashburner, J., Friston, K. J., apr 2011. Diffeomorphic registration using geodesic shooting and GaussNewton optimisation. NeuroImage 55 (3), 954–967.

Avants, B. B., Epstein, C. L., Grossman, M., Gee, J. C., 2008. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. Medical Image Analysis 12 (1), 26–41.

Avants, B. B., Tustison, N. J., Song, G., Cook, P. A., Klein, A., Gee, J. C., 2011. A reproducible evaluation of ANTs similarity metric performance in brain image registration. NeuroImage 54 (3), 2033–2044.

Avants, B. B., Tustison, N. J., Stauffer, M., Song, G., Wu, B., Gee, J. C., 2014. The Insight ToolKit image registration framework. Frontiers in Neuroinformatics 8 (April), 1–13.

Beg, M. F., Miller, M. I., Trouvé, A., Younes, L., feb 2005. Computing Large Deformation Metric Mappings via Geodesic Flows of Diffeomorphisms. International Journal of Computer Vision 61 (2), 139–157.

Bookstein, F. L., 1997. Quadratic variation of deformations. In: Information Processing in Medical Imaging. pp. 15–28.

Burger, M., Modersitzki, J., Ruthotto, L., jan 2013. A Hyperelastic Regularization Energy for Image Registration. SIAM Journal on Scientific Computing 35 (1), B132–B148.

Chen, P., 2011. Hessian Matrix vs. Gauss-Newton Hessian Matrix. Siam 49 (4), 1417–1435.

Christensen, G. E., 1999. Consistent Linear-Elastic Transformations for Image Matching. In: Kuba, A., Šáamal, M., Todd-Pokropek, A. (Eds.), Information Processing in Medical Imaging. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 224–237.

Christensen, G. E., Geng, X., Kuhl, J. G., Bruss, J., Grabowski, T. J., Pirwani, I. A., Vannier, M. W., Allen, J. S., Damasio, H., 2006. Introduction to the Non-rigid Image Registration Evaluation Project (NIREP). Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 4057 LNCS, 128–135.

Coalson, T. S., Van Essen, D. C., Glasser, M. F., 2018. The impact of traditional neuroimaging methods on the spatial localization of cortical areas. Proceedings of the National Academy of Sciences 115 (27), E6356–E6365.

Fischl, B., aug 2012. FreeSurfer. NeuroImage 62 (2), 774–781.

Jenkinson, M., Smith, S., 2001. A global optimisation method for robust affine registration of brain images. Medical Image Analysis 5 (2), 143–156.

Klein, A., Andersson, J., Ardekani, B. A., Ashburner, J., Avants, B., Chiang, M. C., Christensen, G. E., Collins, D. L., Gee, J., Hellier, P., Song, J. H., Jenkinson, M., Lepage, C., Rueckert, D., Thompson, P., Vercauteren, T., Woods, R. P., Mann, J. J., Parsey, R. V., 2009. Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. NeuroImage 46 (3), 786–802.

Klein, S., Staring, M., Murphy, K., Viergever, M., Pluim, J., jan 2010. elastix: A Toolbox for Intensity-Based Medical Image Registration. IEEE Transactions on Medical Imaging 29 (1), 196–205.

Knutsen, A. K., Chang, Y. V., Grimm, C. M., Phan, L., Taber, L. A., Bayly, P. V., 2010. A New Method to Measure Cortical Growth in the Developing Brain. Journal of Biomechanical Engineering 132 (10), 101004.

Miller, M., Banerjee, A., Christensen, G., Joshi, S., Khaneja, N., Grenander, U., Matejic, L., 2003. Statistical methods in computational anatomy. Statistical Methods in Medical Research 6 (3), 267–299.

Miller, M. I., Christensen, G. E., Amitt, Y., Grenander, U. L. F., 1993. Mathematical textbook of deformable neuroanatomies 90 (December), 11944–11948.

Moller, M. F., jan 1993. A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks 6 (4), 525–533.

NVIDIA, 2019. CUDA C Programming Guide (February).
URL https://docs.nvidia.com/cuda/cuda-c-programming-guide/

Ou, Y., Akbari, H., Bilello, M., Da, X., Davatzikos, C., oct 2014. Comparative Evaluation of Registration Algorithms in Different Brain Databases With Varying Difficulty: Results and Insights. IEEE Transactions on Medical Imaging 33 (10), 2039–2065.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., 2007. Numerical recipes : the art of scientific computing. Cambridge University Press.

Robinson, E. C., Garcia, K., Glasser, M. F., Chen, Z., Coalson, T. S., Makropoulos, A., Bozek, J., Wright, R., Schuh, A., Webster, M., Hutter, J., Price, A., Cordero Grande, L., Hughes, E., Tusor, N., Bayly, P. V., Van Essen, D. C., Smith, S. M., Edwards, A. D., Hajnal, J., Jenkinson, M., Glocker, B., Rueckert, D., feb 2018. Multimodal surface matching with higher-order smoothness constraints. NeuroImage 167 (September 2017), 453–465.

Robinson, E. C., Jbabdi, S., Glasser, M. F., Andersson, J., Burgess, G. C., Harms, M. P., Smith, S. M., Van Essen, D. C., Jenkinson, M., oct 2014. MSM: A new flexible framework for Multimodal Surface Matching. NeuroImage 100, 414–426.

Rohlfing, T., feb 2012. Image Similarity and Tissue Overlaps as Surrogates for Image Registration Accuracy: Widely Used but Unreliable. IEEE Transactions on Medical Imaging 31 (2), 153–163.

Rueckert, D., Sonoda, L. I., Hayes, C., Hill, D. L. G., Leach, M. O., Hawkes, D. J., 1999. Nonrigid registration using free-form deformations: application to breast MR images. IEEE Transactions on Medical Imaging 18 (8), 18:712—-21.

Ruthotto, L., Kugel, H., Olesch, J., Fischer, B., Modersitzki, J., Burger, M., Wolters, C. H., sep 2012. Diffeomorphic susceptibility artifact correction of diffusion-weighted magnetic resonance images. Physics in Medicine and Biology 57 (18), 5715–5731.

Schnabel, J. A., Rueckert, D., Quist, M., Blackall, J. M., Castellano-Smith, A. D., Hartkens, T., Penney, G. P., Hall, W. A., Liu, H., Truwit, C. L., Gerritsen, F. A., Hill, D. L. G., Hawkes, D. J., 2001. A Generic Framework for Non-rigid Registration Based on Non-uniform Multi-level Free-Form Deformations. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Vol. 2208. pp. 573–581.

Van Essen, D. C., jan 1997. A tension-based theory of morphogenesis and compact wiring in the central nervous system. Nature 385 (6614), 313–318.

Vercauteren, T., Pennec, X., Perchant, A., Ayache, N., 2009. Diffeomorphic demons: efficient non-parametric image registration. NeuroImage 45 (1 Suppl), S61–S72.