Fast Diffeomorphic Image Registration via Fourier-Approximated Lie Algebras

Article *in* International Journal of Computer Vision · January 2019

DOI: 10.1007/s11263-018-1099-x

2 authors, including:

Miaomiao Zhang
University of Virginia
40 PUBLICATIONS **504** CITATIONS

International Journal of Computer Vision
https://doi.org/10.1007/s11263-018-1099-x

Fast Diffeomorphic Image Registration via Fourier-Approximated Lie Algebras

Miaomiao Zhang[1] · **P. Thomas Fletcher**[2]

Received: 27 October 2016 / Accepted: 4 May 2018

Abstract
This paper introduces Fourier-approximated Lie algebras for shooting (FLASH), a fast geodesic shooting algorithm for diffeomorphic image registration. We approximate the infinite-dimensional Lie algebra of smooth vector fields, i.e., the tangent space at the identity of the diffeomorphism group, with a low-dimensional, bandlimited space. We show that most of the computations for geodesic shooting can be carried out entirely in this low-dimensional space. Our algorithm results in dramatic savings in time and memory over traditional large-deformation diffeomorphic metric mapping algorithms, which require dense spatial discretizations of vector fields. To validate the effectiveness of FLASH, we run pairwise image registration on both 2D synthetic data and real 3D brain images and compare with the state-of-the-art geodesic shooting methods. Experimental results show that our algorithm dramatically reduces the computational cost and memory footprint of diffemorphic image registration with little or no loss of accuracy.

Keywords Fourier-approximated Lie algebras · Geodesic shooting · Diffeomorphic image registration

1 Introduction

The goal of diffeomorphic image registration is to find a smooth mapping, which also has a smooth inverse, between two images. Diffeomorphic image registration is an important component in many medical image analysis tasks, for instance, atlas-based image segmentation and statistical shape analysis. Previous work computed diffeomorphic transformations by integrating over time-dependent velocity fields, using Lagrange transport equations (Dupuis and UGrenander, Miller MI, 1998; Trouvé 1998). Beg et al. (2005) later developed the large-deformation diffeomorphic metric mapping (LDDMM) framework, which introduced a distance metric in the space of diffeomorphisms. In this setting, diffeomorphic image registration is formulated as a variational problem expressing the optimal solution as a geodesic flow in the space of diffeomorphisms. Having such a distance metric builds a solid foundation for basic statistical analysis of diffeomorphisms, including least-squares atlas building as a minimization of the sum-of-squared residual distances to data (Joshi et al. 2004), anatomical shape variability quantification (Zhang and Fletcher 2014), groupwise comparison (Vaillant et al. 2004), and geodesic regression (Niethammer et al. 2011; Singh et al. 2013).

However, the computational cost and the memory consumption of LDDMM are prohibitively high on dense image grids, especially for large imaging studies. The algorithm employs a gradient descent scheme that requires expensive numerical solutions to partial differential equations on the flow of time-varying velocity fields and diffeomorphisms (Beg et al. 2005). Moreover, the gradient decent optimization often gets stuck in local minima due to the high-dimensional parameterization of the deformation fields, and the convergence can be quite slow. To address these issues, Vialard et al. (2012) developed a geodesic shooting algorithm, which estimates only the initial velocity field at time point zero. Geodesic shooting is able to greatly simplify the parameterization of a geodesic, based on the fact that a geodesic path of diffeomorphisms is uniquely determined by the initial conditions of the geodesic evolution equations.

Communicated by Daniel Cremers.

✉ Miaomiao Zhang
miaomiao@cse.lehigh.edu

P. Thomas Fletcher
fletcher@sci.utah.edu

[1] Lehigh University, 113 research drive, Bethlehem, PA, USA

[2] University of Utah, 72 S central campus drive, Salt Lake City, UT, USA

Published online: 15 May 2018

Although the geodesic shooting algorithm avoids storing the entire time-dependent velocity fields at each iteration, the computations needed for gradient evaluations are still expensive on a high-dimensional dense grid.

One approach to alleviate the cost of diffeomorphic image registration is representing the diffeomorphisms by stationary velocity fields, i.e., velocity fields that remain constant in time (Arsigny et al. 2006). A set of ordinary differential equations (ODEs) is solved to generate the flow of diffeomorphisms. In contrast to the original LDDMM, this one-parameter subgroup parameterization is not based on a variational principle, but enables fast estimation of diffeomorphisms as shown in (Ashburner 2007). The solutions are obtained efficiently by composing successive diffeomorphisms over time using scaling and squaring combined with a multiresolution strategy. Other approaches to fast diffeomorphic registration include the greedy algorithm (Christensen et al. 1996) and the log demons algorithm (Vercauteren et al. 2009). These methods iteratively apply gradient updates to a single deformation field rather than a full time-dependent flow. While these methods are much faster and more memory efficient in exploring a large solution space, they (1) do not minimize a global variational problem and (2) do not provide geodesic paths or any type of distance metric between images that is critical to statistical shape analysis.

In this paper, we develop a novel model that dramatically speeds up diffeomorphic image registration while nicely preserving the distance metric. In contrast to the previous work that has discretized smooth velocity fields on dense spatial grids, we are the first to propose a discrete approximation of velocity fields with a finite set of Fourier frequencies. The key insight of our work is that the smooth velocity fields associated with diffeomorphisms are well-approximated in a bandlimited space with much lower dimensions. Furthermore, most computations of the geodesic evolution equations and adjoint Jacobi field equations needed for gradient descent methods can be carried out completely in the Fourier domain and thus remain in this low-dimensional, band-limited representation. This paper is an extension of Zhang and Fletcher (2015) with two major differences: (1) we incorporate the computational complexity and convergence analysis to investigate the efficiency of FLASH at a deep level and (2) we provide more in-depth derivations of our algorithm. We also mention the difference between our work and the discrete parameterization of diffeomorphisms by a finite set of control points (Durrleman et al. 2011), where a similar dimensionality reduction is achieved for the forward geodesic evolution of the control points. However, the gradient terms in image matching still requires computation of adjoint systems on dense image grids, while our FLASH algorithm is able to compute the adjoint system in the low-dimensional representation. To demonstrate the effectiveness of our model, we test a pairwise registration on both 2D synthetic data and 3D brain images and compare its convergence, run-time, and memory consumption with the leading LDDMM method. Experimental results show that our FLASH algorithm is an order of magnitude faster than the state-of-the-art, while converging to better solutions, i.e., lower values of the registration objective function.

2 Background

In this section, we briefly review the mathematical background for LDDMM and geodesic shooting.

2.1 Space of Diffeomorphisms

Let $\Omega = \mathbb{R}^d/\mathbb{Z}^d$ be a d-dimensional torus domain with periodic boundary conditions. A *diffeomorphism* of Ω is a bijective, C^∞ mapping $\phi : \Omega \to \Omega$, with C^∞ inverse ϕ^{-1}. The space of diffeomorphisms is an infinite-dimensional *Lie group*, denoted by $\mathrm{Diff}(\Omega)$. The *Lie algebra* associated with $\mathrm{Diff}(\Omega)$ consists of all C^∞ vector fields on Ω. This Lie algebra forms the tangent space $V = T_{\mathrm{id}}\mathrm{Diff}(\Omega)$ at the identity $\mathrm{id} \in \mathrm{Diff}(\Omega)$, equipped with a binary operator called the *Lie bracket*. For any two vector fields $v, w \in V$, the Lie bracket is defined by

$$[v, w] = Dv \cdot w - Dw \cdot v, \tag{1}$$

where D is the first derivative operator, and $\cdot$ denotes pointwise matrix-vector multiplication. The negative of the Lie bracket forms an ad operator on two vector fields, e.g., $\mathrm{ad}_v w = -[v, w]$.

We consider time-varying diffeomorphisms, $\phi(t, x) : [0, 1] \times \Omega \to \Omega$, which are generated as flows of time-varying velocity fields, $v(t, x) : [0, 1] \times \Omega \to \mathbb{R}^d$. To simplify the notation, we use subscripts for the time variable, i.e., $\phi_t(x) = \phi(t, x)$ and $v_t(x) = v(t, x)$, where $t \in [0, 1]$ and $x \in \Omega$. Given a time-varying velocity field, we generate a time-varying diffeomorphism as the flow $t \mapsto \phi_t \in \mathrm{Diff}(\Omega)$ (see Fig. 1) by integrating an ordinary differential equation (ODE)

$$\frac{d\phi_t}{dt} = v_t \circ \phi_t. \tag{2}$$

2.2 Metrics on Diffeomorphisms

A metric measures the distance between two diffeomorphisms and forms the mathematical foundation for statistical analysis, such as mean estimation, variability quantification, and regression. To define such a distance function in the space of diffeomorphisms, we use a weak *Riemannian metric* that

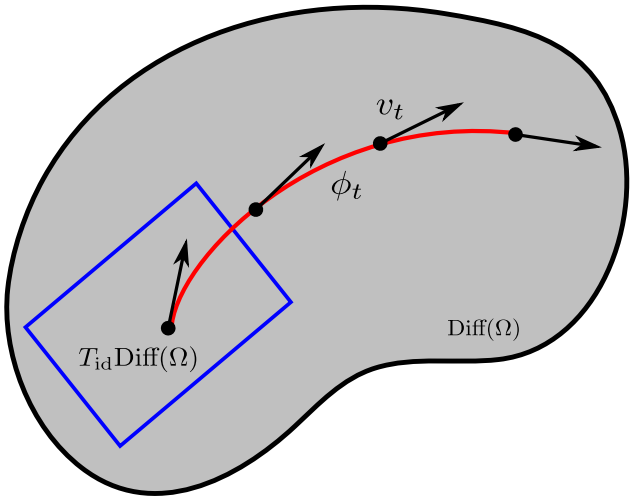

Fig. 1 Flow of diffeomorphisms

is a family of inner products between any two tangent vectors $v, w \in V$ in the Lie algebra, i.e.,

$$\langle v, w\rangle_V = \int_\Omega \langle Lv(x), w(x)\rangle\, dx,$$

with a symmetric, positive-defi K is the inverse of L. For velocities v, w that live on the tangent space of any other element $\phi \in \mathrm{Diff}(\Omega)$, we define the inner product by pulling back the velocities to the identity under a *right-invariant metric* as

$$\langle v, w\rangle = \langle v \circ \phi^{-1}, w \circ \phi^{-1}\rangle_V. \tag{3}$$

The distance function between a diffeomorphism ϕ and the identity id is then defined as an integral over the norm of time-dependent velocity fields, i.e.,

$$\mathrm{dist}(\mathrm{id}, \phi) = \int_0^1 \|v_t\|_V\, dt.$$

A geodesic curve with fixed end point is characterized by an extremum of the energy function $\int_0^1 \|v_t\|_V^2\, dt$ that satisfies the Euler-Poincaré differential (EPDiff) equation (Arnol'd 1966; Miller et al. 2006)

$$\frac{\partial v_t}{\partial t} = -\mathrm{ad}^\dagger_{v_t} v_t = -K\left[(Dv_t)^T m_t + Dm_t\, v_t + m_t\, \mathrm{div}\, v_t\right], \tag{4}$$

where $\mathrm{ad}^\dagger$ is the adjoint of the ad operator, and div denotes the divergence. This process is known as *geodesic shooting*, stating that the geodesic path $\{\phi_t\}$ can be uniquely determined by integrating a given initial velocity $v_0 \in V$ at $t = 0$ forward in time using the rule (4).

2.3 LDDMM with Geodesic Shooting

Consider images as square-integrable functions defined on Ω, that is, an image is an element of $L^2(\Omega, \mathbb{R})$. The problem of diffeomorphic image registration on the source image I_0 and the target image I_1 is to minimize an energy function of a sum-of-squared distance (also known as image matching) with a regularization term

$$E(v_t) = \frac{1}{2\sigma^2}\|I_0 \circ \phi_1^{-1} - I_1\|^2_{L^2} + \int_0^1 \langle Lv_t, v_t\rangle\, dt, \tag{5}$$

where $\circ$ is an interpolation operator that resamples I_0 by the smooth mapping ϕ^{-1} and σ^2 represents image noise variance.

The geodesic shooting algorithm allows to estimate only the initial velocity v_0 instead of the entire sequence of functions v_t (Younes et al. 2009; Vialard et al. 2012). We rewrite the optimization of (5) equivalently as

$$E(v_0) = \frac{1}{2\sigma^2}\|I_0 \circ \phi_1^{-1} - I_1\|^2 + \langle Lv_0, v_0\rangle, \quad \text{s.t. Eq. (4).} \tag{6}$$

Vialard et al. (2012) proved that the solution of (6) is $v_0 = Km_0$, where m_0 is the initial momentum represented as an estimated scalar field P multiplied by the gradient of the source image I_0, i.e., $m_0 = \nabla I_0 P$. However, optimizing over the scalar momentum P has a disadvantage while extending it to an atlas building framework: the estimation of the atlas and the scalar field is coupled, which leads to a poor convergence. To resolve this issue, Singh et al. (2013) proposed to optimize in the vector space of initial momenta m_0 that decouples the observed image data from diffeomorphisms, resulting in a better convergence rate and numerical stability.

3 Fourier-Approximated Lie Algebras

In this section, we introduce a Fourier-approximated Lie algebra and corresponding approximated Lie bracket that provide a discrete representation of velocity fields.

A key observation is that the velocity fields in the geodesic evolution equation, also known as the EPDiff Eq. (4), stay in a low frequency domain. The K operator in (4) is a low-pass filter, and as such, suppresses high frequencies in the velocity fields (see Fig. 2). This indicates that the velocity fields are approximately bandlimited to a certain maximum frequency. Previous implementations of geodesic shooting on a full-dimensional image grid compute the high frequency components that end up being forced close to zero by K. In contrast, we propose to develop a low-dimensional discretization of the velocity fields as bandlimited signals in the frequency domain.

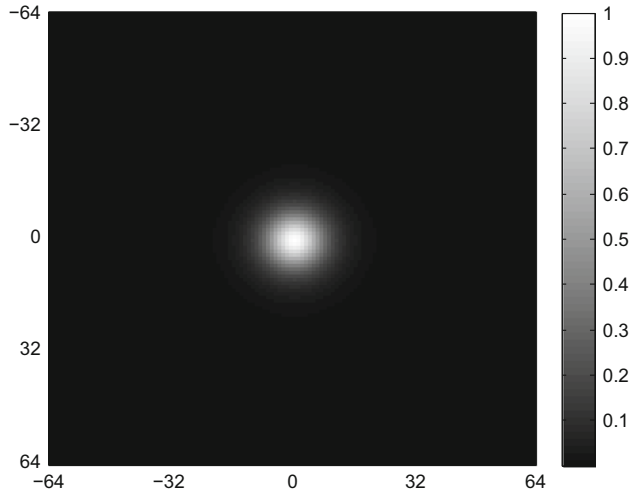

Fig. 2 Fourier coefficients of the discretized K operator on a N^2 grid ($N = 128$), with parameters $\alpha = 3$, $c = 3$

3.1 Space of Bandlimited Velocity Fields and Metrics

Let $\tilde{V}$ denote the space of real-valued velocity fields on Ω that have a finite number of non-zero Fourier coefficients. We define the frequency bounds $n_1, \ldots, n_d$ in each of the dimensions of Ω. Any element $\tilde{v} \in \tilde{V}$ is a multidimensional array: $\tilde{v}_{k_1,\ldots,k_d} \in \mathbb{C}^d$, where $k_i \in \{0, \ldots, n_i - 1\}$ with $i \in \{1, \ldots, d\}$ is the frequency index along the ith axis. In order to ensure $\tilde{v}$ represents a real-valued vector field in the spatial domain, we have the constraint that $\tilde{v}_{k_1,\ldots,k_d} = \tilde{v}^*_{n_1-k_1,\ldots,n_d-k_d}$, where $*$ denotes the complex conjugate.

There is a natural projection mapping, $\nu : V \to \tilde{V}$, of V into the space $\tilde{V}$ of complex vector fields, given by the Fourier series expansion

$$\nu(v)(k) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} v_{x_1,\ldots,x_d} e^{-j2\pi k_1 x_1} \cdots e^{-j2\pi k_d x_d} \, dx_1 \ldots, dx_d, \tag{7}$$

where $k = (k_1, \ldots, k_d)$, $j^2 = -1$, and $x = (x_1, \ldots, x_d)$ is the spatial index.

Vice versa, the inclusion mapping $\iota : \tilde{V} \to V$, of $\tilde{V}$ into the space V is

$$\iota(\tilde{v})(x) = \sum_{k_1=0}^{n_1-1} \cdots \sum_{k_d=0}^{n_d-1} \tilde{v}_{k_1,\ldots,k_d} e^{j2\pi k_1 x_1} \cdots e^{j2\pi k_d x_d}.$$

By using a multi-index notation, we write a simple expression of the inclusion mapping as

$$\iota(\tilde{v})(x) = \sum_{0 \leqslant k < n} \tilde{v}(k) e^{j2\pi kx}. \tag{8}$$

The inner product at identity $\tilde{e}$ on $\tilde{V}$ is given by the Fourier coefficients of the L operator as

$$\langle \tilde{v}, \tilde{w} \rangle_{\tilde{V}} = \sum_{0 \leqslant k < n} \langle \tilde{L}\tilde{v}(k), \tilde{w}(k) \rangle,$$

where $\tilde{L} : \tilde{V} \to \tilde{V}^*$ maps a tangent vector $\tilde{v}$ in the frequency domain to its dual momentum vector $\tilde{m} = \tilde{L}\tilde{v}$, which can be represented as a diagonal matrix of the frequency of $L = (-\alpha\Delta + I)^c$. For a d-dimensional image grid, the Fourier coefficient $\tilde{L}_{k_1,\ldots,k_d}$ of a Laplace operator discretized with the central difference method is

$$\tilde{L}_{k_1,\ldots,k_d} = \left[-2\alpha \left(\cos \frac{2\pi k_1}{N_1} + \cdots + \cos \frac{2\pi k_d}{N_d} - d \right) + 1 \right]^c,$$

with its inverse operator

$$\tilde{K}_{k_1,\ldots,k_d} = 1/\tilde{L}_{k_1,\ldots,k_d}. \tag{9}$$

To define a right-invariant metric at any other point ϕ, we map the velocity fields back to a full spatial domain by (7) first, and then pull back by the right composition with ϕ^{-1} to identity using (3).

3.2 Fourier-Based Lie Bracket

We now define a discrete representation of the continuous Lie bracket in (1).

Definition 1 For any two vector fields $\tilde{v}, \tilde{w} \in \tilde{V}$, a finite-dimensional Lie bracket in the discrete Fourier domain is

$$[\tilde{v}, \tilde{w}] = (\tilde{D}\tilde{v}) * \tilde{w} - (\tilde{D}\tilde{w}) * \tilde{v}, \tag{10}$$

where $\tilde{D}\tilde{v}$ is a tensor product $\tilde{\eta} \otimes \tilde{v}$ with $\tilde{\eta}_{k_1,\ldots,k_d} = (j \sin(2\pi k_1), \ldots, j \sin(2\pi k_d))$ representing the Fourier frequencies of a central difference Jacobian matrix D, and $*$ is a truncated convolution operator.[1]

Since the pointwise multiplication of two vector fields in the spatial domain corresponds to the convolution in the Fourier domain, we can easily decompose the multiplication of a square matrix and vector field in the Fourier domain as a single convolution for each row of the matrix. For notational simplicity, we denote all matrix-vector field and vector-vector field convolutions as $*$ [see (18) in Appendix A for an explicit formulation of $*$].

The vector space $\tilde{V}$, when equipped with the bracket operation (10), is a finite-dimensional approximated Lie algebra.

[1] Since a circular convolution between two bandlimited signals does not preserve the bandlimit, we define a convolution operation on zero-padded signals by truncating the output back to the bandlimits, n_i, in each dimension to guarantee the Lie bracket is closed.

For all vector fields $\tilde{u},\ \tilde{v},\ \tilde{w} \in \tilde{V}$ and $a,\ b \in \mathbb{R}$, the following properties of the linearity and anticommutativity are satisfied:

- Linearity: $[a\tilde{u} + b\tilde{v},\ \tilde{w}] = a\,[\tilde{u},\ \tilde{w}] + b\,[\tilde{v},\ \tilde{w}]$.
- Anticommutativity: $[\tilde{u},\ \tilde{v}] = -[\tilde{v},\ \tilde{u}]$.

However, it is difficult to preserve the Jacobi identity

$$[\tilde{u},\ [\tilde{v},\ \tilde{w}]] + [\tilde{w},\ [\tilde{u},\ \tilde{v}]] + [\tilde{v},\ [\tilde{w},\ \tilde{u}]] = 0$$

that has been mistakenly proved in (Zhang and Fletcher 2015). The product rule in a discrete derivative in a finite difference scheme introduces extra terms at the end and thus break the conservation of the Jacobi identity, e.g.,

$$\begin{aligned}\tilde{D}(\tilde{u} * \tilde{v}) = \ & \tilde{D}\tilde{u} * \tilde{v} \\ & + \tilde{u} * \tilde{D}\tilde{v} + (\tilde{D}^{+}\tilde{u} * \tilde{D}^{+}\tilde{v} - \tilde{D}^{-}\tilde{u} * \tilde{D}^{-}\tilde{v})\end{aligned}$$

for one-dimensional central difference scheme, where $\tilde{D}^{+}$ denotes the forward difference and $\tilde{D}^{-}$ is the backward difference. In this paper, we approximate the Lie algebra in a finite-dimensional space and show its efficiency for estimating diffeomorphisms.

3.3 Geodesic Shooting in Finite-Dimensional Spaces

Analogous to (4), we define a geodesic evolution equation in the discrete Fourier domain as

$$\frac{\partial \tilde{v}_t}{\partial t} = -\mathrm{ad}^{\dagger}_{\tilde{v}_t}\,\tilde{v}_t = -\tilde{K}\left[(\tilde{D}\tilde{v}_t)^{T} \star \tilde{m}_t + \tilde{\Gamma}(\tilde{m}_t \otimes \tilde{v}_t)\right], \quad (11)$$

where $\star$ denotes a truncated auto-correlation and $\tilde{\Gamma}$ is a discrete divergence of a vector field $\tilde{v}$ computed as a sum over the frequencies of central difference along each dimension, i.e., $\tilde{\Gamma}\tilde{v} = \sum_{0 \leqslant k < n} \tilde{v}(k)\tilde{\eta}^{T}(k)$. Note that applying the $\tilde{K}$ operator on bandlimited velocity fields can be easily computed as an element-wise multiplication of a diagonal matrix with the Fourier coefficients defined in (9). Details of deriving this EPDiff equation in the Fourier domain are in Appendix A.

Using the inclusion mapping $\iota : \tilde{V} \to V$ defined in (8), a time sequence of bandlimited velocity fields in $\tilde{V}$ consequently generates a flow of diffeomorphisms, $t \mapsto \phi_t \in \mathrm{Diff}(\Omega)$, in the following way

$$\frac{d\phi_t}{dt} = \iota(\tilde{v}_t) \circ \phi_t, \quad (12)$$

and its inverse mapping ϕ_t^{-1} is computed through

$$\frac{d\phi_t^{-1}}{dt} = -D\phi_t^{-1} \cdot \iota(\tilde{v}_t). \quad (13)$$

4 Fast Diffeomorphic Image Registration

We present a geodesic shooting algorithm for diffeomorphic image registration by employing a gradient decent scheme in the low-dimensional approximated Lie algebras. All computations of geodesic shooting are in the reduced low-dimensional space, producing a time-varying velocity, $t \mapsto \tilde{v}_t \in \tilde{V}$ that generates a flow of diffeomorphic transformations. We parameterize the diffeomorphisms by the low-dimensional velocity $\tilde{v}_0$ and write the original LDDMM energy function (6) as

$$E(\tilde{v}_0) = \frac{1}{2\sigma^2}\left\| I_0 \circ \phi_1^{-1} - I_1 \right\|^2 + \langle \tilde{L}\tilde{v}_0, \tilde{v}_0 \rangle, \quad \text{s.t. Eq. (11).} \quad (14)$$

Computations of the gradient of (14) require two parts: forward integration of $\tilde{v}_0$ by (11) to compute the gradient $\nabla_{\tilde{v}_1} E$ at time point $t = 1$ followed by integrating backward to the initial point $t = 0$.

Similar to Beg et al. (2005), $\nabla_{\tilde{v}_1} E$ is obtained after using the projection mapping $\nu : V \to \tilde{V}$ in (7) as

$$\nabla_{\tilde{v}_1} E = \nu\left(-K\left(\frac{1}{\sigma^2}(I_0 \circ \phi_1^{-1} - I_1) \cdot \nabla(I_0 \circ \phi_1^{-1})\right)\right). \quad (15)$$

We next introduce reduced adjoint Jacobi fields in the bandlimited velocity space to integrate the gradient (15) at $t = 1$ backward to $t = 0$.

4.1 Reduced Adjoint Jacobi Fields in Bandlimited Velocity Spaces

Before we develop the adjoint Jacobi field in the low-dimensional Lie algebra, we first briefly review its definition in the continuous space.

Consider a variation of geodesics $\gamma(s, t) : (-\epsilon, \epsilon) \times [0, 1] \to \mathrm{Diff}(\Omega)$, with $\gamma(s, 0) = \mathrm{id}$ and $\gamma(0, t) = \phi_t$. Such a variation corresponds to a variation of the initial velocity $\frac{d\gamma}{dt}(s, t)|_{t=0} = v_0 + s\delta v_0$ that produces a "fan" of geodesics (illustrated in Fig. 3), where δ denotes a variation on v_0. Taking the derivative of this variation results in a Jacobi field $J(t) = \frac{d\gamma}{ds}(s, t)|_{s=0}$ with its adjoint operator, called adjoint Jacobi field.

In this paper, we employ a reduced version of adjoint Jacobi fields (Bullo 1995; Hinkle et al. 2014) that equivalently carries all the computations in the Lie algebra. In contrast to the current estimation of LDDMM methods (Beg et al. 2005; Singh et al. 2013; Vialard et al. 2012), this reduced adjoint Jacobi fields completely decouples the integration of smooth velocity fields from noisy images, which

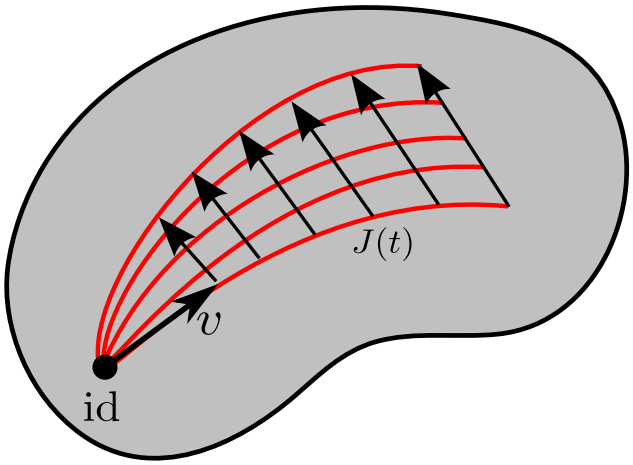

Fig. 3 Jacobi fields

substantially improves the stability and convergence of the optimization.

Because we are using a right-invariant metric on Diff(Ω), it is natural to express the Jacobi field $J(t)$ in the Lie algebra using right invariance. We define such a vector field, $U(t) \in \tilde{V}$, as the right-invariant Jacobi field $U(t) = \nu\left(J(t) \circ \phi_t^{-1}\right)$. Similarly, we define a variation of the right-reduced velocity $\tilde{v}$ as $\delta\tilde{v}$. These variables satisfy the following reduced Jacobi equations:

$$\frac{d}{dt}\begin{pmatrix} U \\ \delta\tilde{v} \end{pmatrix} = \begin{pmatrix} \mathrm{ad}_{\tilde{v}} & I \\ 0 & \mathrm{sym}_{\tilde{v}} \end{pmatrix}\begin{pmatrix} U \\ \delta\tilde{v} \end{pmatrix}, \tag{16}$$

where $\mathrm{sym}_{\tilde{v}}\delta\hat{v} = -\mathrm{ad}^{\dagger}_{\tilde{v}}\delta\hat{v} - \mathrm{ad}^{\dagger}_{\delta\hat{v}}\tilde{v}$.

To transport the gradient term $\nabla_{\tilde{v}_1}E$ backward to the space of initial velocity fields, we use reduced Jacobi fields that are simply computed by the adjoint of the reduced Jacobi equations in (16). This results in another set of ordinary differential equations (ODE) as

$$\frac{d}{dt}\begin{pmatrix} \hat{U} \\ \delta\hat{v} \end{pmatrix} = \begin{pmatrix} -\mathrm{ad}^{\dagger}_{\tilde{v}} & 0 \\ -I & -\mathrm{sym}^{\dagger}_{\tilde{v}} \end{pmatrix}\begin{pmatrix} \hat{U} \\ \delta\hat{v} \end{pmatrix}, \quad \text{or}$$
$$\frac{d\hat{U}}{dt} = -\mathrm{ad}^{\dagger}_{\tilde{v}}\hat{U}, \quad \frac{d\delta\hat{v}}{dt} = -\hat{U} - \mathrm{sym}^{\dagger}_{\tilde{v}}\delta\hat{v}, \tag{17}$$

where $\hat{U}$, $\delta\hat{v} \in \tilde{V}$ are introduced auxiliary adjoint variables and $\mathrm{sym}^{\dagger}_{\tilde{v}}\delta\hat{v} = \mathrm{ad}^{\dagger}_{\delta\hat{v}}\tilde{v} - \mathrm{ad}_{\tilde{v}}\delta\hat{v}$. For more details on the derivation of the reduced adjoint Jacobi field equations, see Bullo (1995).

Given the initial conditions $\hat{U}(1) = \nabla_{\tilde{v}_1}E$ and $\delta\hat{v}(1) = 0$ at time point $t = 1$, we obtain the transported gradient $\delta\hat{v}(0)$ by integrating the adjoint ODE (17) backward in time to $t = 0$.

Finally, we arrive at the gradient of E w.r.t. $\tilde{v}_0$ as

$$\nabla_{\tilde{v}_0}E = \tilde{v}_0 + \delta\hat{v}(0).$$

To summarize, the gradient descent algorithm that minimizes the energy function (14) is listed in Algorithm 1. Note that the results of our paper are produced by Euler integration to solve differential equations. Alternative integrators such as Euler, Runge–Kutta, or other high-order methods could also be applied.

Algorithm 1: FLASH

Input: source image I_0, target image I_1.
Initialize $\tilde{v}_0 = 0$.
repeat
(a) **Forward shooting of $\tilde{v}_0$:** forward integrate the geodesic evolution equation (11) to generate $\{\tilde{v}_t\}$ at discrete time points.
(b) **Compute the inverse diffeomorphism ϕ^{-1}:** integrate the negative velocity fields backward in time by using (13).
(c) **Compute gradient at time point $t = 1$:** compute the gradient $\nabla_{\tilde{v}_1}E$ by (15).
(d) **Compute gradient at time point $t = 0$:** integrate the reduced adjoint Jacobi field equations (17) backward in time to obtain the gradient $\nabla_{\tilde{v}_0}E$.
(e) Update $\tilde{v}_0 \leftarrow \tilde{v}_0 - \varepsilon\nabla_{\tilde{v}_0}E$, where ε is the step size.
until convergence

4.2 Complexity Analysis

We briefly review the gradient descent algorithm for vector momenta LDDMM with geodesic shooting (Singh et al. 2013) and compare its complexity with the proposed FLASH algorithm. The optimization of the vector momenta LDDMM method is done by minimizing an augmented energy function with introduced time-dependent adjoint variables. The related gradient term is computed in the dual space of the initial velocity fields, which is the space of initial momenta m_0, by adding Lagrange multipliers: (a) the geodesic constraint that comes from the EPDiff equation (4), (b) the image transport equation along time t, and (c) the enforcement of $m_t = Lv_t$ to constrain the diffeomorphism $\{\phi_t\}$ to be smooth. The detailed steps are summarized in Algorithm 2.

Table 1 demonstrates a comparison of the computational complexity and memory requirements between FLASH and vector momenta LDDMM on pairwise 3D image registration. The computation of forward and backward shooting of velocity fields [step (a) and (d)] is dramatically lower in FLASH than vector momenta LDDMM. Moreover, our model obtains the inverse of diffeomorphism ϕ_t^{-1} directly from integrating the negative bandlimited velocity fields backwards under the left invariant metric of diffeomorphisms, without wasting the computational cost of solving ϕ_t [step (b)].

Table 1 Computational complexity of FLASH and vector momenta LDDMM (T: number of integration time steps; d: image dimension; n: number of low frequencies along each dimension; N: number of image voxels along each dimension)

	Complexity		Memory	
	FLASH	LDDMM	FLASH	LDDMM
Forward shooting	$O(Tn^d \log n)$	$O(TN^d \log N)$	$O(Tn^d)$	$O(TN^d)$
Compute ϕ^{-1}	$O\left(TN^d\right)$	$O\left(TN^d\right)$	$O(N^d)$	$O(TN^d)$
Compute gradient at $t = 1$	$O\left(N^d \log N\right)$	$O\left(N^d\right)$	$O(N^d)$	$O(N^d)$
Compute gradient at $t = 0$	$O(Tn^d \log n)$	$O(TN^d \log N)$	$O(n^d)$	$O(N^d)$

Algorithm 2: Vector momenta LDDMM

Input: source image I_0, target image I_1.
Initialize $m_0 = 0$.
repeat
(a) **Forward shooting of v_0:** forward integrate the geodesic evolution equations (4) on a dense grid to generate $\{v_t\}$ at each discrete time point.
(b) **Compute the inverse diffeomorphism ϕ^{-1}:** compute and store the diffeomorphism ϕ_t by integrating (2) followed by solving its inverse ϕ_t^{-1} using the fixed-point iteration method.
(c) **Compute gradient at time point $t = 1$:** compute the gradient, $\nabla_{m_1} E$, of the energy (6) at $t = 1$.
(d) **Compute gradient at time point $t = 0$:** integrate the adjoint equations in a high-dimensional image space to update $\nabla_{m_0} E$.
(e) Update $m_0 \leftarrow m_0 - \varepsilon \nabla_{m_0} E$.
until convergence

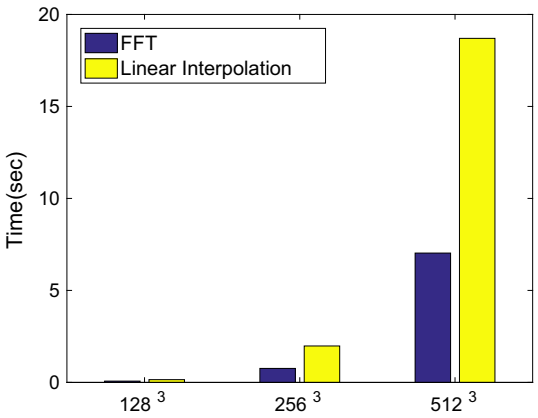

Fig. 4 Exact run-time comparison between Fourier transform and grid interpolation for different N

While the complexity of computing the inverse diffeomorphism ϕ^{-1} [step (b)] in vector momenta LDDMM is the same as our algorithm, the exact run-time is an order of magnitude more expensive in practice due to the unspecified constant factors in the big-O notations (as shown later in Table 2). The empirical running time of the most expensive operation fast Fourier transformation in FLASH and linear interpolation in vector momenta LDDMM with different scales of image dimension in Fig. 4 shows that the linear interpolation costs twice as much as the Fourier transform.

5 Results

To evaluate our proposed FLASH method, we perform pairwise image registrations at different levels of truncated dimension. We compare the final energy defined in (6), run-time, memory consumption, and convergence rate of our method versus the open source implementation of vector momenta LDDMM (https://bitbucket.org/scicompanat/vectormomentum) (Singh et al. 2013). For a fair comparison, we set $\alpha = 3.0, c = 3.0, \sigma = 0.03$ with $T = 10$ time-steps for Euler integration in all experiments. We emphasize that we used the same full-dimensional registration energy from (5) for all runs so that all experimental results are comparable.

5.1 Data

We validate all methods on both 2D synthetic data and real 3D MRI brain data. We generate a set of 128^2 binary images of sine waves with different frequencies $f = 2, 4, \ldots, 16$. The real brain MRI scans are 60 healthy subjects randomly selected from the OASIS database across a broad age range of 60 to 95 (Marcus et al. 2007). The MR images are of dimension 128^3, $1 \times 1 \times 1$ mm^3 isotropic voxels and underwent skull-stripping, downsampling, intensity normalization to the range [0, 1], and co-registration with affine transformations.

5.2 Experiments

5.2.1 2D Synthetic Data

We run pairwise image registration on 2D synthetic images with the truncated dimension $n = 32$, at which point the Fourier coefficients of the K operator have decayed to nearly zero (shown in Fig. 2). Figure 5 demonstrates that FLASH is able to achieve the same image match in a much lower dimensional space ($n = 32$) as it is using the full-dimensional space ($n = 128$). Furthermore, FLASH arrives at a better image match than vector momenta LDDMM as the frequency of the sine waves increases. However, both methods have difficulty matching the fine details in the sine waves at frequency

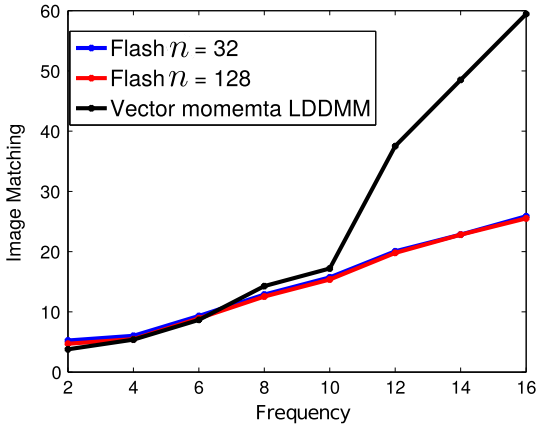

Fig. 5 Comparison of image matching with different number of frequencies $f = 2, 4, \ldots, 16$ between FLASH ($n = 32, 128$) and vector momenta LDDMM

$f = 16$ (see Fig. 6). This indicates that this level of detail is finer than what can be captured using these particular parameters in the regularization (i.e., α and c in the L operator).

5.2.2 3D Brain MRI

We first test pairwise image registration with FLASH at different levels of truncated dimension $n = 4, 8, 16, 32, 64, 128$.

Figure 7 shows that FLASH achieves a lower total energy than vector momenta LDDMM at the truncated dimensions $n \geq 16$ with much less time and memory. Since an increased dimension beyond $n = 16$ does not improve the image registration energy, we conclude that $n = 16$ is sufficient to capture the transformations between images. Our proposed method arrives at the optimal solution in 1.96 s per iteration and 168.4 MB memory. In comparison, vector momenta LDDMM requires 45.806 s per iteration and 1708.1 MB memory, which is even more expensive than a full-dimensional FLASH (30.1 s per iteration and 505.35 MB memory when $n = 128$).

Table 2 gives an exact run-time for each step required for a gradient decent iteration in FLASH ($n = 16$) and in vector momenta LDDMM. Our FLASH method is approximately 23 times faster than the leading LDDMM method.

Figure 8 demonstrates the convergence graph of total energy, image matching, and velocity energy by FLASH at dimension $n = 16, 128$. It shows that our method converges to an optimal solution with lower function energy and runs faster than vector momenta LDDMM. The performance of FLASH with $n = 16$ is fairly close to the full dimension $n = 128$, which indicates that our model does not lose any key information in a bandlimited space.

Atlas building We next demonstrate the benefits of our algorithm in building an atlas image (mean image) (Joshi et al. 2004) for large data analysis. We initialize the atlas as the average of image intensities and set the truncated dimension as $n = 16$. We use a message passing interface (MPI) parallel programming implementation for both our model and vector momenta LDDMM and scatter each image onto an individual processor. With 100 iterations for gradi-

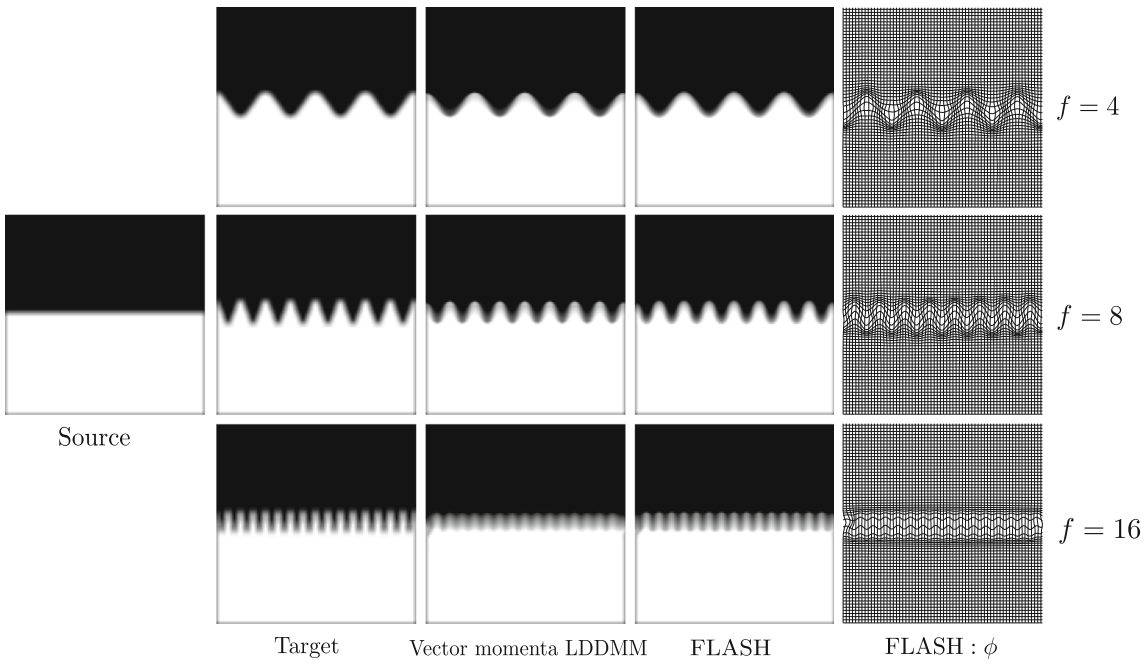

Fig. 6 Left to right: source, target, deformed source by vector momenta LDDMM, FLASH with $n = 32$ and the transformations. Top to bottom: sine waves with different frequencies $f = 4, 8, 16$

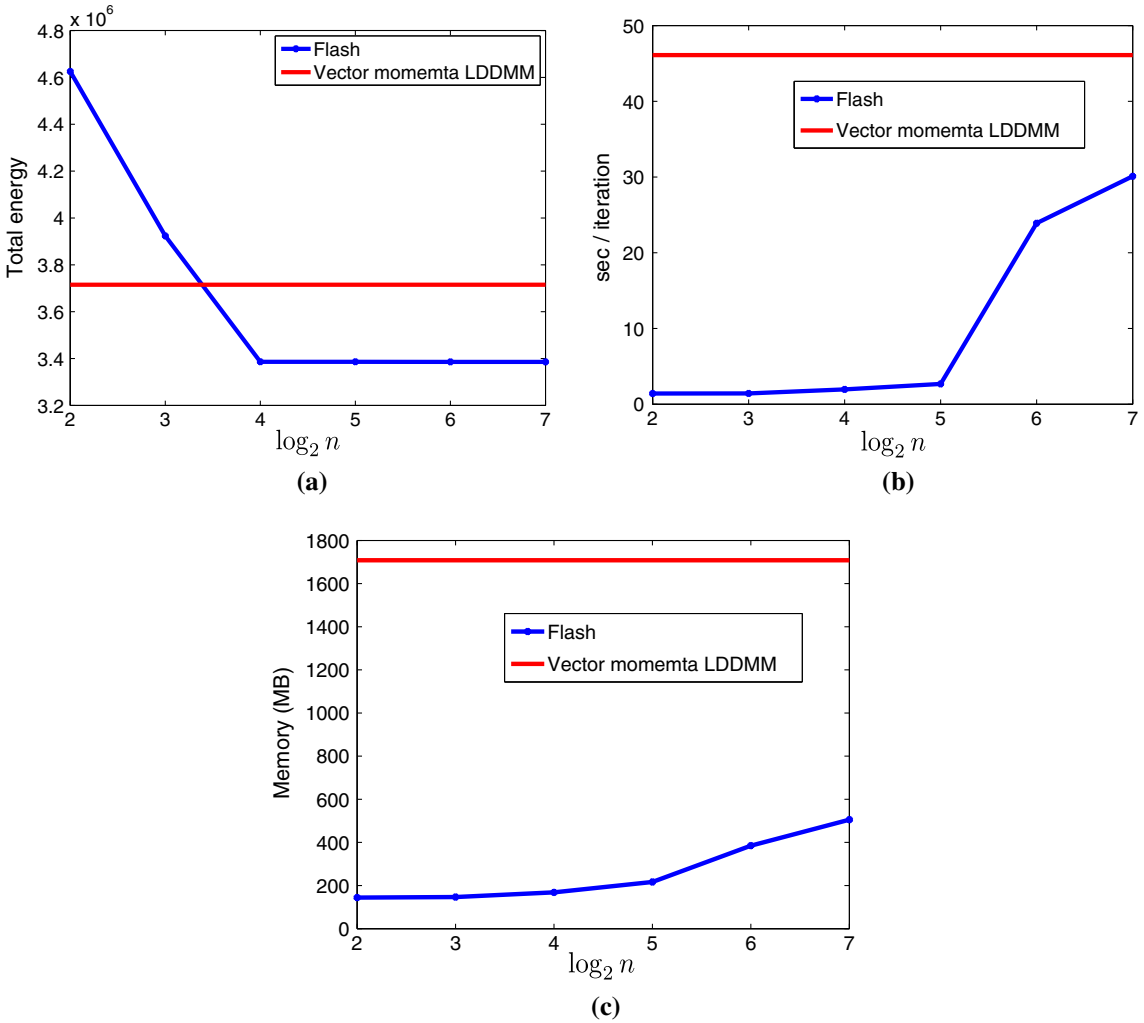

Fig. 7 Comparison of **a** total energy, **b** time consumption, and **c** memory requirement

ent descent, FLASH builds the atlas in 6.3 min, whereas the vector momenta LDDMM requires 2 h.

The left side of Fig. 9 shows the axial and coronal slices from 8 of the selected 3D MRI dataset. The right side shows the atlas image estimated by FLASH, followed by the atlas estimated by vector momenta LDDMM. We see from the difference image between the two atlas results that FLASH generates a very similar atlas to the vector momenta LDDMM, but at a fraction of the time and memory cost.

6 Conclusion and Future Work

We presented a fast geodesic shooting algorithm FLASH for diffeomorphic image registration. Our method is the first to introduce a definition of Fourier-approximated Lie algebras that provides a discrete representation of the tangent space of diffeomorphisms in a bandlimited velocity space. Another key contribution of our approach is that we can compute the geodesic evolution equations, as well as the adjoint Jacobi field equations required for gradient descent methods, entirely in a low-dimensional bandlimited space. This gives us a dramatically fast diffeomorphic image registration algorithm without loss of accuracy. This work can be used for further statistical analysis (for example, anatomical shape variation) since we preserved the distance metric on the space of diffeomorphisms.

There are several interesting directions for future work based on FLASH. First of all, in this paper, we pre-specify the parameters of noise variance, regularization, and the truncated dimension. Since different parameters could lead to a different solution for the diffeomorphic transformation, we can develop a Bayesian formulation much like Zhang et al. (2013) to estimate all these parameters automatically from the data. This approach would require more expensive computations to treat diffeomorphisms as latent variables and integrate them out from the target distribution, but our algo-

Table 2 Comparison with vector momenta LDDMM for exact run-time on 128^3 images with $n = 16$

	Run-time	
	FLASH (s)	LDDMM (s)
Forward shooting	0.1	9.25
Compute ϕ^{-1}	1.3	20.5
Compute $\nabla_{\tilde{v}_1} E$	0.16	0.056
Compute $\nabla_{\tilde{v}_0} E$	0.40	16

rithm has the ability to make inference via Monte Carlo sampling of the hidden diffeomorphisms more feasible.

Secondly, FLASH paves the way for efficient computations in large statistical studies by using LDDMM. Other speed-up strategies, for instance, a second-order Gauss-Newton step, similar to the one proposed in Ashburner and Friston (2011), or a multiresolution optimization scheme, could easily be added on top of our algorithm for further speed improvement.

Finally, we note that the code for FLASH is available as a free library online: (https://bitbucket.org/FlashC/flashc) and it has been successfully used in (Hromatka et al. 2015; Zhang et al. 2016, 2017).

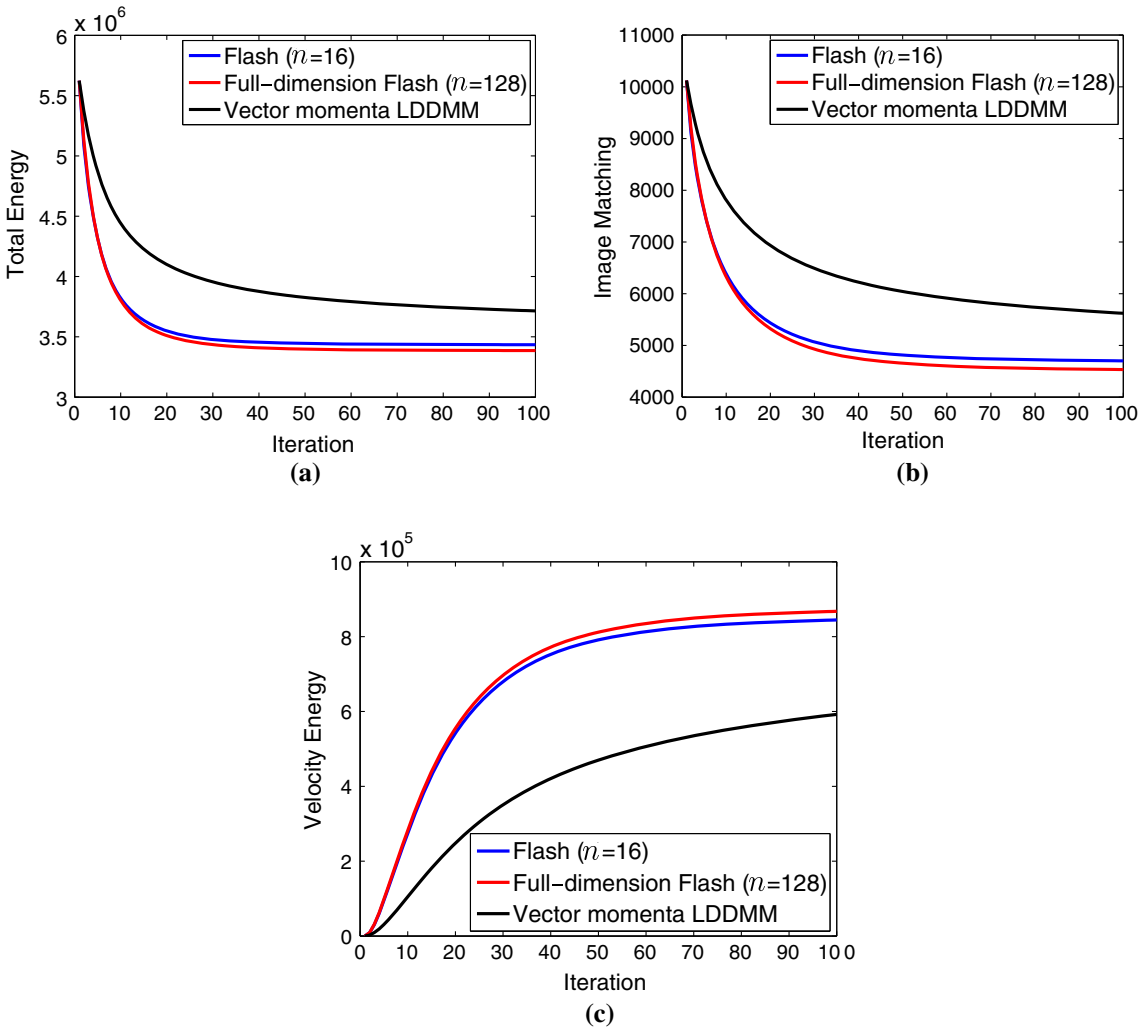

Fig. 8 Comparison of convergence on **a** total energy, **b** image matching, and **c** velocity energy

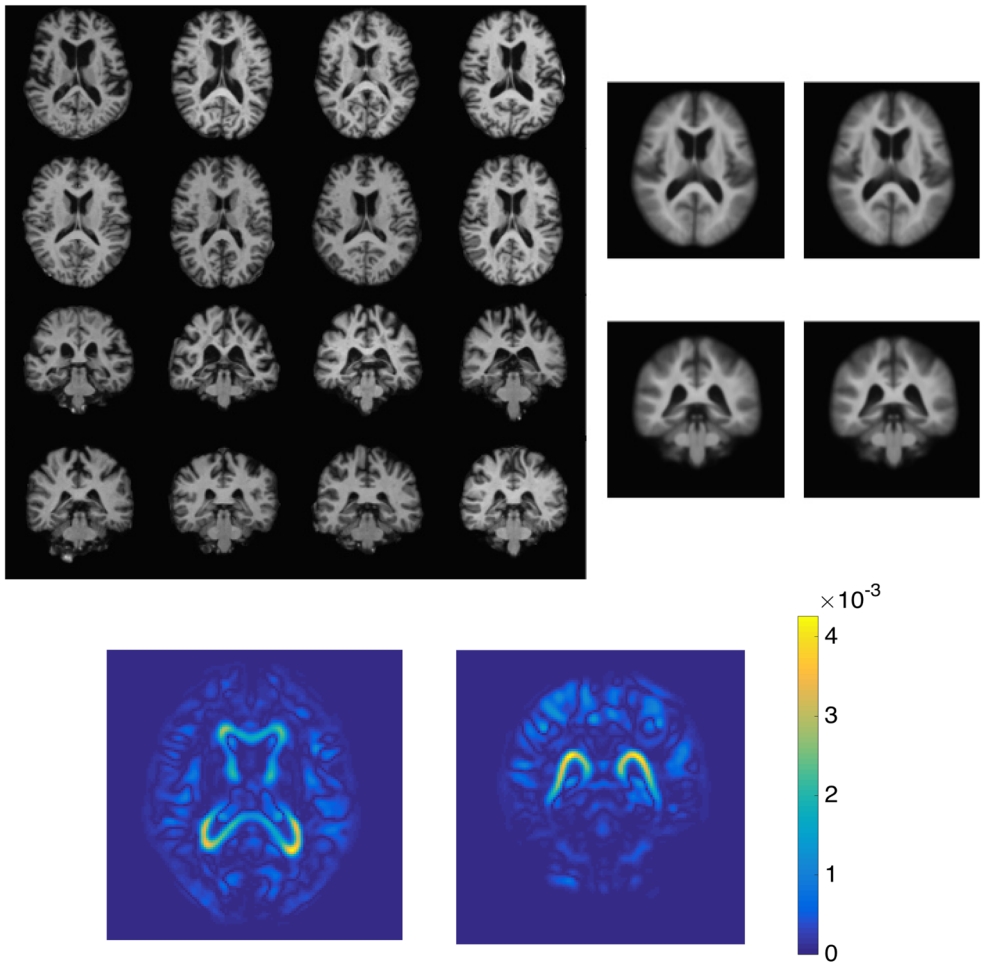

Fig. 9 Top left: axial and coronal slices from 8 of the input 3D MRIs. Middle to right: atlas estimated by our model with truncated dimension $n = 16$ and vector momenta LDDMM. Bottom: axial and coronal view of atlas intensity difference

Acknowledgements This work was supported by NIH Grant 5R01EB007688 and NSF CAREER Grant 1054057.

Appendix A

Properties of truncated convolution

We compute the truncated convolution in a bandlimited space by padding a sufficient number of zeros, $n_i - 1$, at the end of each dimension, and then truncate the signal back to its original space. Note that (1) the convolution is modulo-q_i ($q_i = 2n_i - 1$) circular convolution, (2) the zero frequency component is shifted to the center of the domain.

We first introduce the kth element of a truncated convolution of any tangent vector field $\tilde{v}, \tilde{w} \in \tilde{V}$ as

$$[\tilde{v} * \tilde{w}]_k = \sum_{0 \leqslant l < q} \tilde{v}_{k-l} \tilde{w}_l, \tag{18}$$

where $l = (l_1, \ldots, l_d) \in \mathbb{Z}^d$ are the padded frequency indices with $l_i \in \{0, \ldots, q_i\}$ along the ith axis. Let $M_i = \lfloor \frac{n_i - 1}{2} \rfloor$, then $k = (k_1, \ldots, k_d) \in \mathbb{Z}^d$ denotes truncated frequency indexes with $k_i \in \{M_i, \ldots, M_i + n_i - 1\}$. Notice here the $k - l$ represents $k - l \pmod q$ with cyclic boundary conditions, and the vector field $\tilde{v}$ can also be replaced with a matrix field.

We then prove the commutativity and associativity of this truncated convolution. For any $\tilde{u}, \tilde{v}, \tilde{w} \in \tilde{V}$,

- Commutativity: $\tilde{u} * \tilde{v} = \tilde{v} * \tilde{u}$

Proof The kth element of $\tilde{u} * \tilde{v}$ is

$$[\tilde{u} * \tilde{v}]_k = \sum_{0 \leqslant l < q} \tilde{u}_{k-l} \tilde{v}_l.$$

By changing the coordinates using $k - l = k'$ where $k \in [M, M + n - 1]$ and $k' \in [0, q]$ due to the cyclic condition,

we rewrite the equation above as

$$[\tilde{u} * \tilde{v}]_k = \sum_{0 \leqslant k' < q} \tilde{u}_{k'} \tilde{v}_{k-k'} = [\tilde{v} * \tilde{u}]_k.$$

□

– Associativity: $(\tilde{u} * \tilde{v}) * \tilde{w} = \tilde{u} * (\tilde{v} * \tilde{w})$

Proof The kth element of $(\tilde{u} * \tilde{v}) * \tilde{w}$ is

$$[(\tilde{u} * \tilde{v}) * \tilde{w}]_k = \sum_{0 \leqslant l' < q} \sum_{0 \leqslant l < q} \tilde{u}_{k-l'-l} \tilde{v}_l \tilde{w}_{l'},$$

where $k \in [M, M+n-1]$. We then rewrite the equation above by changing the coordinates using $k - l' - l = a$, where $a \in [0, q]$, as

$$[(\tilde{u} * \tilde{v}) * \tilde{w}]_k = \sum_{0 \leqslant l' < q} \sum_{0 \leqslant a < q} \tilde{u}_a \tilde{v}_{k-a-l'} \tilde{w}_{l'} = [\tilde{u} * (\tilde{v} * \tilde{w})]_k.$$

□

Deriving ad* operator

Before showing the derivation of ad* in a bandlimited space, we first introduce the pairing between a result of the truncated convolution (18) and a momentum vector field $\tilde{m} \in \tilde{V}^*$ is

$$\begin{aligned}(\tilde{m}, \tilde{v} * \tilde{w}) &= \sum_{M \leqslant k < M+n} (\tilde{m}_{k-M}, \sum_{0 \leqslant l < q} \tilde{v}_{k-l} \tilde{w}_l) \\ &= \sum_{M \leqslant k < M+n} \sum_{0 \leqslant l < q} (\tilde{m}_{k-M}, \tilde{v}_{k-l} \tilde{w}_l)\end{aligned}$$

We are now ready to derive the ad* operator from its definition

$$(\mathrm{ad}^*_{\tilde{v}} \tilde{m}, \tilde{w}) = (\tilde{m}, \mathrm{ad}_{\tilde{v}} \tilde{w}).$$

Since $\mathrm{ad}_{\tilde{v}} \tilde{w} = [\tilde{v}, \tilde{w}]$, after plugging in the definition of Lie bracket defined in (10) we have

$$\begin{aligned}(\mathrm{ad}^*_{\tilde{v}} \tilde{m}, \tilde{w}) &= \left(\tilde{m}, (\tilde{D}\tilde{v}) * \tilde{w} - (\tilde{D}\tilde{w}) * \tilde{v}\right) \\ &= \left(\tilde{m}, (\tilde{D}\tilde{v}) * \tilde{w}\right) - \left(\tilde{m}, (\tilde{D}\tilde{w}) * \tilde{v}\right) \\ &= \sum_{M \leqslant k < M+n} \sum_{0 \leqslant l < q} (\tilde{m}_{k-M}, \tilde{v}_{k-l} \tilde{\eta}^T_{k-l} \tilde{w}_l) \\ &\quad - (\tilde{m}_{k-M}, \tilde{w}_l \tilde{\eta}^T_l \tilde{v}_{k-l})\end{aligned}$$

To separate $\tilde{w}$, we change coordinates by defining $k - M = k', l - M = l'$ where $k' \in [0, n-1]$ and $l' \in [-M, q-M]$. For the purpose of notation simplicity, we drop the range of variables in the following equations as

$$\begin{aligned}(\mathrm{ad}^*_{\tilde{v}} \tilde{m}, \tilde{w}) &= \sum_{k'} \sum_{l'} (\tilde{m}_{k'}, \tilde{v}_{k'-l'} \tilde{\eta}^T_{k'-l'} \tilde{w}_{l'+M}) \\ &\quad - (\tilde{m}_{k'}, \tilde{w}_{l'+M} \tilde{\eta}^T_{l'+M} \tilde{v}_{k'-l'}) \\ &= \sum_{k'} \sum_{l'} \left((\tilde{v}_{k'-l'} \tilde{\eta}^T_{k'-l'})^{T*} \tilde{m}_{k'}, \tilde{w}_{l'+M}\right) \\ &\quad - (\tilde{v}^*_{k'-l'} \tilde{m}^T_{k'}, \tilde{w}_{l'+M} \tilde{\eta}^T_{l'+M}) \\ &= \sum_{k'} \sum_{l'} \left((\tilde{v}_{k'-l'} \tilde{\eta}^T_{k'-l'})^{T*} \tilde{m}_{k'}, \tilde{w}_{l'+M}\right) \\ &\quad - (\tilde{v}^*_{k'-l'} \tilde{m}^T_{k'} \tilde{\eta}^*_{l'+M}, \tilde{w}_{l'+M}) \\ &= \sum_{k'} \sum_{l'} \left((\tilde{v}_{k'-l'} \tilde{\eta}^T_{k'-l'})^{T*} \tilde{m}_{k'}, \tilde{w}_{l'+M}\right) \\ &\quad + (\tilde{v}^*_{k'-l'} \tilde{m}^T_{k'} \tilde{\eta}_{l'+M}, \tilde{w}_{l'+M}) \\ &= \left((\tilde{D}\tilde{v})^T \star \tilde{m}, \tilde{w}\right) + \left(\tilde{\Gamma}(\tilde{m} \otimes \tilde{v}), \tilde{w}\right) \\ &= \left((\tilde{D}\tilde{v})^T \star \tilde{m} + \tilde{\Gamma}(\tilde{m} \otimes \tilde{v}), \tilde{w}\right). \qquad (19)\end{aligned}$$

Finally, we have

$$\begin{aligned}\mathrm{ad}^*_{\tilde{v}} \tilde{m} &= (\tilde{D}\tilde{v})^T \star \tilde{m} + \tilde{\Gamma}(\tilde{m} \otimes \tilde{v}), \\ \mathrm{ad}^\dagger_{\tilde{v}} \tilde{v} &= \tilde{K} \mathrm{ad}^*_{\tilde{v}} \tilde{m} = \tilde{K}[(\tilde{D}\tilde{v})^T \star \tilde{m} + \tilde{\Gamma}(\tilde{m} \otimes \tilde{v})].\end{aligned}$$

References

Arnol'd, V. I. (1966). Sur la géométrie différentielle des groupes de Lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits. *Ann Inst Fourier*, *16*, 319–361.

Arsigny, V., Commowick, O., Pennec, X., & Ayache, N. (2006). A log-Euclidean framework for statistics on diffeomorphisms. In *Medical image computing and computer-assisted intervention—MICCAI 2006*, Springer (pp. 924–931).

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

Ashburner, J., & Friston, K. J. (2011). Diffeomorphic registration using geodesic shooting and Gauss–Newton optimisation. *NeuroImage*, *55*(3), 954–967.

Beg, M., Miller, M., Trouvé, A., & Younes, L. (2005). Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, *61*(2), 139–157.

Bullo, F. (1995). Invariant affine connections and controllability on Lie groups. Technical Report for Geometric Mechanics, California Institute of Technology.

Christensen, G. E., Rabbitt, R. D., & Miller, M. I. (1996). Deformable templates using large deformation kinematics. *IEEE Transactions on Image Processing*, *5*(10), 1435–1447.

Dupuis, P., Grenander, U., & Miller, M. I. (1998). Variational problems on flows of diffeomorphisms for image matching. *Quarterly of Applied Mathematics*, *56*(3), 587.

Durrleman, S., Prastawa, M., Gerig, G., & Joshi, S. (2011). Optimal data-driven sparse parameterization of diffeomorphisms for population analysis. In *Information processing in medical imaging*, Springer (pp. 123–134).

Hinkle, J., Fletcher, P. T., & Joshi, S. (2014). Intrinsic polynomials for regression on Riemannian manifolds. *Journal of Mathematical Imaging and Vision*, *50*(1–2), 32–52.

Hromatka, M., Zhang, M., Fleishman, G. M., Gutman, B., Jahanshad, N., Thompson, P., & Fletcher, P. T. (2015). A hierarchical Bayesian model for multi-site diffeomorphic image atlases. In *Medical image computing and computer-assisted intervention—MICCAI 2015*, Springer (pp. 372–379).

Joshi, S., Davis, B., Jomier, M., & Gerig, G. (2004). Unbiased diffeomorphic atlas construction for computational anatomy. *NeuroImage*, *223*(Supplement 1), 151–160.

Marcus, D. S., Wang, T. H., Parker, J., Csernansky, J. G., Morris, J. C., & Buckner, R. L. (2007). Open access series of imaging studies (OASIS): Cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, *19*(9), 1498–1507.

Miller, M. I., Trouvé, A., & Younes, L. (2006). Geodesic shooting for computational anatomy. *Journal of Mathematical Imaging and Vision*, *24*(2), 209–228. https://doi.org/10.1007/s10851-005-3624-0.

Niethammer, M., Huang, Y., & Vialard, F. X. (2011). Geodesic regression for image time-series. In *International conference on medical image computing and computer-assisted intervention*, Springer (pp. 655–662).

Singh, N., Hinkle, J., Joshi, S., & Fletcher, P. T. (2013). A vector momenta formulation of diffeomorphisms for improved geodesic regression and atlas construction. In *International symposium on biomedial imaging (ISBI)*.

Trouvé, A. (1998). Diffeomorphisms groups and pattern matching in image analysis. *International Journal of Computer Vision*, *28*(3), 213–221.

Vaillant, M., Miller, M. I., Younes, L., & Trouvé, A. (2004). Statistics on diffeomorphisms via tangent space representations. *NeuroImage*, *23*, S161–S169.

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

Vialard, F. X., Risser, L., Rueckert, D., & Cotter, C. J. (2012). Diffeomorphic 3D image registration via geodesic shooting using an efficient adjoint calculation. *International Journal of Computer Vision*, *97*(2), 229–241.

Younes, L., Arrate, F., & Miller, M. (2009). Evolutions equations in computational anatomy. *NeuroImage*, *45*(1S1), 40–50.

Zhang, M., & Fletcher, P. T. (2014). Bayesian principal geodesic analysis in diffeomorphic image registration. In *Medical image computing and computer-assisted intervention–MICCAI 2014*, Springer (pp. 121–128).

Zhang, M., & Fletcher, P. T. (2015). Finite-dimensional Lie algebras for fast diffeomorphic image registration. In *Information processing in medical imaging*.

Zhang, M., Liao, R., Dalca, A. V., Turk, E. A., Luo, J., Grant, P. E., & Golland, P. (2017). Frequency diffeomorphisms for efficient image registration. In *International conference on information processing in medical imaging*, Springer (pp. 559–570).

Zhang, M., Singh, N., & Fletcher, P. T. (2013). Bayesian estimation of regularization and atlas building in diffeomorphic image registration. J. C. Gee, S. Joshi, K. M. Pohl, W. M. Wells & L. Zöllei (Eds.), Information processing in medical imaging (pp. 37–48). Springer.

Zhang, M., Wells III, W. M., & Golland, P. (2016). Low-dimensional statistics of anatomical variability via compact representation of image deformations. In *International conference on medical image computing and computer-assisted intervention*, Springer (pp. 166–173).