Discrete Mechanics and Optimal Control for Image Registration

Robert I McLachlan
Institute of Fundamental Sciences
Massey University
Palmerston North
New Zealand
r.mclachlan@massey.ac.nz

Stephen Marsland
Institute of Information Sciences & Technology
Massey University
Palmerston North
New Zealand
s.r.marsland@massey.ac.nz

Abstract

Diffeomorphic image registration, where images are aligned using diffeomorphic warps, is a popular subject for research in medical image analysis. We introduce a novel algorithm for computing diffeomorphic warps that fits into the framework of Discrete Mechanics and Optimal Control, a popular choice for optimisation methods in numerical analysis. The result is an algorithm that is many times faster than those considered previously.

1. Introduction

Image registration has received much research over the past few years, not least because of its many applications in medicine. For example, it is useful for removing motion artefacts caused by patient breathing, heartbeat, and patient movement [26], for aligning to an atlas [8], for monitoring disease progression [27], for assisting in disease diagnosis [22], and for measuring anatomical variability between subjects [28]. For further details about these applications see [28], and for a (relatively old) survey of medical image registration see [18]. A more general survey of image registration, highlighting its uses in synthetic aperture radar and other applications is given in [30].

For applications in disease diagnosis and measuring anatomical variability some form of measurement on the space of images is essential, to be able to do statistical analysis of the image warps. This generally requires using diffeomorphic image registration, whereby the choice of image warps that can be used to solve the registration problem are constrained to be diffeomorphisms, i.e., smooth functions that have smooth inverses. There has therefore been recent interest in the use of diffeomorphic deformations (warps) to align medical images. One approach to this problem, known as Computational Anatomy, is to introduce group actions as deformable templates that are warped via the actions of a group onto other images [9]. This work has been fundamental to a large amount of research on aligning images through landmark matching, where corresponding points are defined on a set of images, and diffeomorphic warps used to align them. The corresponding group for image analysis is the full diffeomorphism group, not the volume-preserving subgroup that is used in fluid mechanics.

Under a right-invariant Riemannian metric, it can be shown that the geodesics of the motion of a set of landmarks can be computed as an optimisation problem; see [23, 22] for an overview. In this paper we introduce a novel formulation for the problem that is based on implicitly solving the partial differential equations that govern the motion. These partial differential equations are the Euler equations for the full diffeomorphism group, given by equations 1 and 2, for derivations see [23, 14], following [1]. We introduce a particle method that enables us to use the framework of Discrete Mechanics and Optimal Control (DMOC) —see section 3 —to solve for the diffeomorphism directly. This results in an algorithm that is orders of magnitude faster than previous ones.

2. Problem Formulation

We begin by defining the problem of diffeomorphic image registration:

> Assume that there is a diffeomorphism ϕ that takes an image T to a reference image R, i.e., $R = T \circ \phi$. The aim of diffeomorphic image registration is to discover ϕ.

The diffeomorphism ϕ is defined on some domain $\Omega \in \mathbb{R}^2$ or $\mathbb{R}^3$, and the images are typically greyscale, so that $R, T : \mathbb{R}^2 \rightarrow \mathbb{R}$ or $R, T : \mathbb{R}^3 \rightarrow \mathbb{R}$. The method used to find the desired ϕ is generally optimisation of some norm $\|R - T \circ \phi\|$. Typical choices include the L^2 norm (sum-of-squares error) and mutual information [29, 17], although

there are other alternatives, including the correlation ratio [25] and the normalised gradient-bsed method in [10].

In this paper we describe a novel method of constructing the diffeomorphisms. The standard approach in the literature is to use an energy minimisation, which produces the diffeomorphism as a geodesic. There are effectively three different approaches: (a) considering the problem as one of inexact matching on the boundary values (the start and end points of the landmarks) [15, 4]; (b) considering the problem as one of inexact matching using the initial values of position and momentum [5]; (c) considering the problem as one of exact matching [22]. The first approach optimises the warp over the end points of the spline, which are not guaranteed to be reached precisely, the second optimises the warp over the initial momenta of the particles, while the third approach can consider either method equally.

For the case of the full diffeomorphism group, $\mathfrak{G} = \mathrm{Diff}(\mathbb{R}^n)$, that we consider here, the Euler equations are (see [13, 24] for further details):

$$\dot{m} + u \cdot \nabla m + \nabla u^T \cdot m + m(\mathrm{div}\, u) = 0, \qquad (1)$$

where $\dot{m}$ denotes differentiation with respect to time, $u(x,t)$ ($u, x \in \mathbb{R}^n$, $t \in \mathbb{R}$) is a velocity field, and $m(x,t)$ its associated momentum. The velocity u and momentum m are related by:

$$m = \mathcal{A}u, \qquad (2)$$

where $\mathcal{A}$ is an elliptic operator (e.g. $\mathcal{A} = (1 - \nabla^2)^k$) called the inertia operator.

A striking feature of Euler equations on diffeomorphism groups is that they admit (formally, at least) exact solutions in which the momentum is concentrated at a finite set of points that we call particles. For fluid equations these are point vortices, which are widely studied both in their own right and as a means of approximating the evolution of smooth or other vorticities.

For the 2D and 3D Euler fluid equations, convergence of the point vortex solutions to solutions for smooth initial data has been established [3]. The speed of convergence can be improved by smoothing out the point vortices to *vortex blobs* [6], even though the (e.g., Gaussian) blobs are no longer an exact solution of the Euler equations. Instead, their evolution can be regarded as that of delta-functions under a slightly different inertia operator $\mathcal{A}$. The inverse of the inertia operator $\mathcal{A}$ is given by convolution with the Green's function $\mathbf{G}$ of $\mathcal{A}$, i.e., $u = \mathbf{G} * m$, where $*$ denotes convolution and $\mathcal{A}\mathbf{G}(x,x') = \delta(x-x')$ for $x, x' \in \mathbb{R}^n$. We shall only consider rotationally invariant and diagonal $\mathcal{A}$; in this case $\mathbf{G}(x,x') = G(\|x - x'\|)$ for a scalar function G.

3. A Particle Method for Image Registration

We are considering the deformation of an image Ω, with the deformation defined by a set of points i (some subset of the pixels of the image) with position and momentum $(q^i(t), p^i(t))$, where $p_i = \dot{q}_i$ as they move from their initial state (q_0^i, p_0^i) to their endpoint at $t = 1$. Starting from the Euler equations on the diffeomorphism group (1 and 2) we first compute the Hamiltonian, which is the kinetic energy, and then discretise it by introducing the particle ansatz $m(x,t) = \sum_{j=1}^{N} p_j(t)\delta(x - q_j(t))$, where $\delta(\cdot)$ is Kronecker delta function. In analogy with the point vortices fluid dynamics, we call these *point particles*. The evolution of the particles is then given by the following Hamiltonian:

$$H = \frac{1}{2}\sum_{i,j} p_i \cdot p_j G(q_i - q_j), \qquad (3)$$

where $G(\cdot)$ is the Green's function corresponding to the chosen metric on $\mathrm{Diff}(\Omega)$. The most common choice in image registration, and the one that we will use in this paper is the H^∞ metric, which corresponds to using a Gaussian Green's function $G(r) = \frac{1}{\epsilon^2}\exp(-r^2/\epsilon^2)$, where ϵ is the length-scale in the metric. Other choices include the thin-plate spline and clamped-plate spline – see [22] for a review.

Solutions to (1) of this particle form obey Hamilton's equations for (3), in which the components of q_i and p_i are canonically conjugate variables (see [19] for further details). Here $q_1, \ldots, q_N$ represent the positions of the N particles that define the deformation, and $p_1, \ldots, p_N$ their momenta. The equations of motion of the point particles are:

$$\dot{q}_i = \sum_{j=1}^{N} G(\|q_i - q_j\|)p_j \qquad (4)$$

$$\dot{p}_i = -\sum_{j} (p_i \cdot p_j) G'(\|q_i - q_j\|)\frac{q_i - q_j}{\|q_i - q_j\|}. \qquad (5)$$

Computing the diffeomorphism defined by q_i and p_i, $i = 1 \ldots N$ is then simply a case of integrating the motion forward in time using 4 and 5, and then interpolating the motion of the rest of the image in some way. This leads us to our complete algorithm for image registration, following which we discuss several important implementation details. Some examples of 2D registrations using the algorithm are given in the next section. The section references on each line of the algorithm give the section where this is discussed in more detail.

- Choose point particle positions q on image T (Sec. 3.1)
- Initialise the particle momenta p randomly (Sec. 3.2)
- Optimise $\|R - T \circ \phi\|$ over p (Sec. 3.3):
  - For current p, integrate point particles forward in time (Sec. 3.4)
  - Integrate positions of the test particles (Sec. 3.5)

  - Interpolate between the test particles (Sec. 3.5)
  - Compute $\|R - T \circ \phi\|$ for chosen distance measure (Sec. 3.6)
- Add more point particles and iterating (Sec. 3.7)

3.1. Position of point particles

There are several possible choices for placing the point particles. Examples include placing them in a grid, positioning them on points of interest in the image, such as edges and corners, or using the discrepancy image to select places where the two images do not match, based on the objective function. In line with [21], for registration of brains, we initially place some points around the skull of head images, and after optimising them, place more points using the discrepancy image method. For the hand images shown in the next section, we consider both the discrepancy image method and using a uniform grid. In all cases, the momenta were initialised as small random numbers.

It is worth noting that there is no reason why one cannot optimise the point locations as well as their momenta. We have done some initial experiments with this, and it sometimes gives a better solution. Certainly, it can resolve smaller image features, but it is also significantly more prone to getting stuck in poor local minimas, because the optimisation is hugely more complex. Finding suitable implementation methods to get around this problem, possibly using multiple scales of resolution, is one of our current areas of research.

3.2. Initialisation of point momenta

In the current implementation, the momenta of each point particle are initialised with a uniformly random direction, and with a small uniform random magnitude for the warp. One option that does appear to improve the results, although at a moderate computational cost, is to perform a coarse search over this relatively small number of parameters (2 for each of the point particles, of which there may be 10-20 on the initial pass). The question of how to initialise the momenta for additional point particles is discussed in section 3.7.

3.3. Optimisation method

The choice of a suitable optimiser is obviously crucial. In the current implementation we are using the sum-of-squares distance measure, which leads fairly naturally to a least-squares non-linear optimiser. We are currently using the `lsqnonlin` function in Matlab 7.1, which is a subspace trust region method based on the interior-reflective Newton method. Experimentation has found that allowing 100 iterations is usually sufficient for the algorithm to converge reasonably, although further work will investigate this more thoroughly.

3.4. Choice of integrator

The principal component of our method is the computation of the current geodesic, based on q and the current p. This is calculated by numerically integrating the particle dynamics forward in time. The Hamiltonian ordinary differential equations are discretised in time, and then integrated forward. We can choose a timestep for the integration, and the method of numerical integration. The standard choices would generally be Euler's method, or a second-order improvement, such as second-order Runge-Kutta.

The factors that affect the computation of the diffeomorphism are the number of point particles and test particles, the number of timesteps, and the order of the integrator (how errors accumulate during the integration). In consideration of the last two of these points, in section 5 we discuss the possible benefits and disadvantages of using a symplectic integrator. We also discuss a possible reduction in the computational complexity of the algorithm – if there are N point particles and M test particles, the cost of evaluating the vector field is $\mathcal{O}(N^2 + NM)$.

3.5. Test particles and interpolation

We can induce the value of the actual diffeomorphism $\phi(x)$ by the current geodesic by placing particles with zero momentum (so $q(0) = x, p(0) = 0$) at the centre point of each pixel in the image, and computing its trajectory under the induced velocity field (i.e., solving the ODE $\dot{q} = f(q, t)$). These are known as test particles.

In fact, assuming that the deformation is not too large (so that $\|T\phi - 1\|$ is reasonably small), we can make some computational savings by placing a test particle every k pixels, and interpolating ϕ between the test particles. This saves a factor of k^2 computations, but changes the computed diffeomorphism from the exact one that relates to the flow (indeed, it may actually stop the warp being diffeomorphic, although this does not seem to be a problem in general). We have found that using $k = 4$ and bi-linear interpolation has negligible effect on the accuracy in real registrations, as is demonstrated in table 1 in section 4.

3.6. Choice of metric

Inherent in the choice of Green's function $G(r)$ is a choice of the metric under which the particle dynamics occur. There is complete freedom of choice over this metric. By far the most common choice to date for image registration has been to use a Gaussian metric, i.e., Green's function

$$G(r) = \frac{1}{\epsilon^2} \exp(-r^2/\epsilon^2), \tag{6}$$

where ϵ is the length-scale in the metric. The role of this length-scale is important. Clearly, if it is set too small (say smaller than the pixel spacing) then the kernels will not

overlap, and the movement of each particle will be entirely independent of the rest of the image. This will require the number of point particles to tend to infinity to represent an arbitrary diffeomorphism. One way around this problem is to use a function such as the clamped-plate spline [22], which uses a bounded domain with strict boundary conditions to avoid the problem of defining a length-scale; effectively the length-scale is the size of the image. The metric used for the clamped-plate spline is of the form ∇^{2k}. We do not consider the question of how to choose the length-scale in this paper. It may be that starting with a large value of ϵ and allowing it to shrink is a useful method of refining the solution iteratively, but we have not yet looked into this. In the results reported in section 4, a value of $\epsilon = 1$ was used, which is half the width of the image (which is scaled into $[-1, 1]^2$).

The Gaussian is by no means the only possible choice of metric. One fairly general formulation, which includes the Gaussian as the limit as $k \to \infty$, are the H^k metrics, $(1 - \epsilon^2\nabla^2)^k$. Finally, it may well be useful to choose the metric so that it vanishes on some set of motions that are not important. Examples could be affine or rigid motions.

3.7. Adding more points

In our implementation we position new point particles for further levels of optimisation using the discrepancy image. This uses the objective function (here the sum-of-squares error) to find regions where the two images do not match, and then placing new point particles there. Given that, there are two choices of how to initialise the momenta of these new particles: (i) as zero, and (ii) as small random numbers. We have tried both schemes, and found that the first was the most effective. This is not surprising, because points with zero momentum are carried along with the flow, which is a reasonable initial guess for how they should behave, and the optimiser then improves on this.

4. Experiments

We present four experiments in this paper. The first considers how far apart the spacing should be between the test particles. As was discussed in section 3.5, the wider apart they are spaced, the faster the implementation, but the less accurate the results. In order to decide a suitable spacing, we took a series of 10 registrations of hands, as used for the registration shown in figure 3 and described below, and tested out different spacings between the test particles for two different numbers of point particles (with the initial values for the momenta of the point particles fixed between the runs). The average results over the 10 registrations are shown in table 1, and show that a spacing of 4 between test particles provides a reasonable compromise between computational time and final function value, hence we have used a spacing of 4 for all the computations used in this paper.

Figure 1. The initial discrepancy between the two hands to be registered. The final result can be seen in figures 1 and 3.

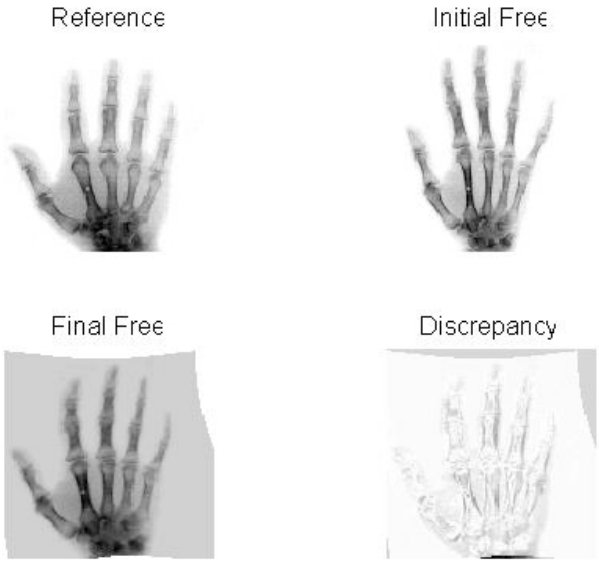

Figure 3. The registration of the two hands. The initial pair are shown in the top row, with the final version of the free image shown on the bottom left, together with the discrepancy image. It can be seen that the registration has been successful.

For the second experiment, we investigated how the performance of the integrators change as the number of timesteps is varied. Table 2 shows the results for registrations of the two hands. The times shown are for a 1.5GHz Apple Powerbook G4. It can be seen that the number of timesteps does not appear to significantly affect the results of the integration.

Finally, we present two different image registrations. The first is of a pair of hand images, while the second are two 2D T1-weighted MR scans of the human brain. Figure 1 shows the initial discrepancy between the two hands to be registered, while figure 3 shows the images and the fi-

Figure 2. The registration of the two hands. The initial pair are shown in the top row, with the final version of the free image shown on the bottom left, together with the discrepancy image. It can be seen that the registration has been successful.

Figure 4. Chequer-board plots showing the different between the initial images of the hands (left), the final images (centre), and the change between the initial and final versions of the free image (right).

Table 1. Comparison of changing the spacing between the test particles. Results are the average of 10 values. A spacing of 4 appears to give a reasonable compromise between computational cost and the final diffeomorphism.

Spacing	Time (s)	Final function value
9 point particles		
12	39.13	7.78e7
8	43.13	7.43e7
4	47.71	7.08e7
1	112.6	6.95e7
25 point particles		
12	123.17	1.06e8
8	133.95	9.82e7
4	163.05	9.37e7
1	944.72	8.85e7

nal discrepancy. These results were computed using 9 knot-points, positioned in a 3×3 grid on the image. The optimiser ran for 40 iterations before converging, and then an additional 7 points were added to the image using the method described in [21], which adds points where the error is currently large. Another 37 iterations were then performed by the optimiser, with the final result being that shown. Figure 2 shows the positions of the points and the initial momenta on the reference image, the final output, and the effect of the warp on a regular grid. Figure 4 provides a different way to interpret the results, showing a chequer-board overlay of the two images before and after the registration, as well as the change. It can be seen that even after this relatively small amount of computation, the registration is very good. Computing this registration took 251.4 seconds on a 1.8GHz G5 Apple Macintosh.

Figure 5 show a sample registration of 2 brains. A set of 10 points were positioned evenly around the skull, and the result optimised for 20 iterations. Following this, an

Table 2. Changing the number timesteps does not appear to significantly affect the results of the registration, but using more timesteps fastly increases the computational cost.

	Euler		RK2	
Timesteps	Func Val	Time	Func Val	Time
9 point particles				
1	6.72e6	61.64	6.88e6	65.81
2	6.82e6	66.38	6.92e6	74.48
4	6.87e6	74.74	6.93e6	91.91
8	6.87e6	92.05	6.96e6	126.50
16	6.86e6	128.46	6.96e6	193.91
25 point particles				
1	5.97e6	186.59	6.33e6	217.76
2	6.02e6	222.07	6.35e6	284.80
4	5.86e6	292.61	6.43e6	412.43
8	6.01e6	417.34	6.32e6	666.75
16	6.13e6	696.1	6.17e6	1258.5

additional set of 11 knotpoints, with 50 iterations of optimisation then being performed. This registration took under 7 minutes on the same computer, and it can be see that the final result is not too bad. There is still work to be done on the interior (and further optimisations do indeed correct this), but the skull and major structures have all been brought into alignment.

5. Use of Symplectic Integrators

The equations of motion (4, 5) are Hamiltonian and their flow is therefore symplectic [11]. In long-time simulations of Hamiltonian systems (in celestial and molecular mechanics, for example) it has been found extremely advantageous to use symplectic integrators, which preserve the symplectic structure. This leads to good energy behaviour and a lack of dissipation. Therefore it is natural to consider their use here; it is also in accord with the 'Discrete Mechanics and Optimal Control' philosophy in which both the cost function and dynamics are discretized in a parallel, Hamiltonian way [16]. In fact, some implementations of image registration by diffeomorphisms have used symplectic integrators, because calculating geodesics by minimizing a discrete path length does give a symplectic integrator [22]. However, the diffeomorphism itself, calculated from the motion of the test particles, has never been done symplectically. We give a preliminary analysis of the cost and benefits of using a symplectic integrator in image registration.

At first sight, the cost is a problem. The cheapest, explicit symplectic integrators apply to separable Hamiltonians of the form $T(p)+V(q)$; Eq. (3) is not separable. Only implicit symplectic integrators, notably the Gaussian Runge-Kuttas [11], are available. These involve solving a set of equations for s internal stages; when $s = 1$, we have the midpoint rule

$$x_{k+1} = x_k + \Delta t f(\bar{x}_k), \quad \bar{x}_k = (x_k + x_{k+1})/2.$$

Moreover, to ensure exact symplecticity and that the solution varies smoothly with respect to the initial conditions, the equations must be solved extremely accurately, generally down to round-off error. In most situations, it is best to simply solve the equations by iteration

$$x_{k+1}^{l+1} = x_k + \Delta t f((x_k + x_{k+1}^l)/2), \quad l = 0, 1, 2, \ldots$$

after choosing some initial guess x_{k+1}^0. If m iterations are required then the cost per time step is ms times the cost of Euler's method. In initial value problems with a large time step, as we want to use here, m can be quite large, say 5–15.

In fact, this cost penalty for initial value problems vanishes for optimization problems, in which we want to repeatedly solve the same initial value problem for a sequence of nearby initial values. We simply store the internal stage values as part of the orbit segment and use this as initial guesses (e.g. x_{k+1}^0 for the midpoint rule) when the initial conditions are changed. Most optimization algorithms estimate the derivatives of the objective function using finite differences, which requires repeatedly altering the initial conditions by about 10^{-6}; for these evaluations we can solve the implicit equations in a *single* iteration. The error constants of the Gaussian Runge-Kutta methods are extremely small so we expect that this method could be superior both for cost and accuracy. The situation is similar to that of boundary value problems, for which Gaussian Runge-Kutta methods are already the method of choice [2], even for general (non-Hamiltonian) first-order systems.

For very large numbers of point particles, the cost $\mathcal{O}(N^2 + NM)$ of evaluating the vector field may be too expensive. The cost can be reduced to $\mathcal{O}(N+M)$ using the marker-and-cell method [12], while still using symplectic integrators for the particle paths [7]. A regular grid with $\mathcal{O}(N)$ grid points is laid over the domain and the particle momenta interpolated to the grid. Then the velocity field induced by the momentum field is calculated on the grid using a fast algorithm, such as multigrid ($\mathcal{O}(N)$) or Fourier transforms ($\mathcal{O}(N \log N)$). This velocity field is interpolated back to the particle positions which are then updated. This algorithm has been implemented with enormous numbers (more than 1 million) particles in an initial value problem in atmospheric dynamics [7]. However, very large numbers of point particles, which may well be required for convergence to an arbitrary diffeomorphism, will introduce new difficulties for the optimization has the problem has now become ill-posed. Some degree of regularization, enforcing smoothness of the initial momentum field, will be required.

As well as cost benefits, does the symplectic condition itself yield superior numerical properties for this problem? It

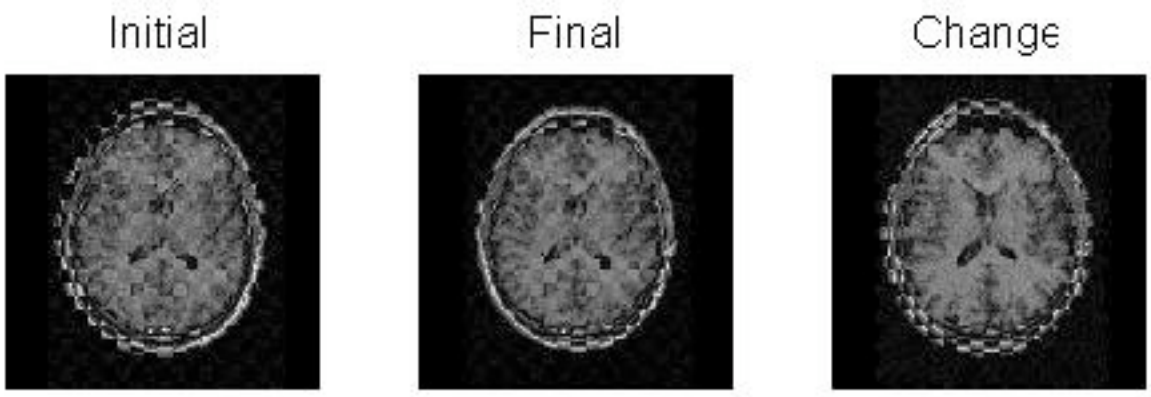

Figure 5. Chequer-board plots showing the different between the initial images of the T1-weighted brains(left), the final images (centre), and the change between the initial and final versions of the free image (right). The registration has lined up the skulls and the major structures within the brain, but there is still more fine-scale work to be done.

is known that it does for optimization problems in spacecraft mission design ([20]) where the integration time is long, the displacements are large, and the errors have a chance to build up. In image registration the integration time is small and is held fixed; integration errors do not accumulate much, so their favourable accumulation by symplectic integrators is less relevant. On the other hand, our experiments indicate (see Figs. ?) that the error of the integrator is a factor in the quality of the registration. Moreover, the symplectic condition specifically constrains the variation of the final particle positions with respect to changes in the initial conditions, which is exactly what we are studying in the optimization problem.

Apart from symplecticity, a major benefit of the Gaussian Runge-Kutta methods in other applications is their unconditional stability for linear problems. This often allows the use of a larger time-step, which is exactly what we want to achieve here.

A critical part of the whole algorithm is to generate diffeomorphisms. For the midpoint rule, this diffeomorphism is given by $x_0 \mapsto x_1$, where $x_1 = x_0 + \Delta t f(\bar{q}, \bar{p}, \bar{x})$. Therefore, if the midpoint rule always generates diffeomorphisms of phase space, in this application it will generate a diffeomorphism of physical space. However, this is not the case. Applied to the 1-dimensional linear test problem $\dot{x} = \lambda x$, the midpoint rule fails to produce a diffeomorphism when $\lambda \Delta t = \pm 2$. (Euler's method fails when $\lambda \Delta t = -1$).

To sum up, the cost considerations of using symplectic Gaussian Runge-Kutta methods in image registration look very promising, and these alone may turn out to support their use. The benefits of the symplectic condition itself are less clear; we have given several positive and negative factors, whose relative importance we plan to determine via experimentation.

6. Conclusions and Open Questions

We have presented a method of performing diffeomorphic image registration using the methods of discrete mechanics and optimal control. The implementation described in this paper has been demonstrated to perform high quality registrations in reasonably short computational time – orders of magnitude less than using energy minimisation methods. It is still a matter for debate whether diffeomorphic methods are suitable for general image registration. However, for applications where it is *variation* that is of interest, for example in disease diagnosis or measurement of anatomical variability, the access to a right-invariant Riemannian metric on the diffeomorphism group makes diffeomorphic registration methods essential.

There are a great many unanswered questions and areas for future research. We are particularly interested in the dynamical behaviour of the Euler equations on the diffeomorphism group, and how it relates to point vortices in fluid dynamics, which act on the volume-preserving subgroups. We have papers in preparation on precisely this topic.

However, with regard to using the method for image registration, there are also several areas for further work. Firstly, we are currently investigating the use of the midpoint rule symplectic integrator, as discussed above, and a second question that we highlighted earlier in the paper is that of a suitable choice of metric. In this paper we have used the Gaussian metric, which is equivalent to $H^k, k \to \infty$. These H^k metrics have the form $(1 - \epsilon^2 \nabla^2)^k$, which has an inherent length-scale ϵ. The role of both the metric and the length-scale need further investigation – there is no guarantee that the Gaussian and any chosen length-scale are the correct choices for any particular problem, and some methods of comparing the results of using different metrics on a set of different problems is currently under investigation.

Acknowledgments

We are grateful to Jan Modersitzki for the use of the hand images.

References

[1] V. I. Arnold. Sur la géométrie différentielle des groupes de Lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits. *Annales de L'Institut Fourier (Grenoble)*, 16(1):319–361, 1966. 1

[2] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell. *Numerical solution of boundary value problems for ordinary differential equations*. SIAM, Philadelphia, 1995. 6

[3] J. T. Beale and A. Majda. Vortex methods. II: Higher order accuracy in two and three dimensions. *Mathematics of Computation*, 39(159):29 – 52, 1982. 2

[4] M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, February 2005. 2

[5] V. Camion and L. Younes. Geodesic interpolating splines. In M. Figueiredo, J. Zerubia, and A. K. Jain, editors, *Proceedings of EMMCVPR'01*, volume 2134 of *Lecture Notes in Computer Science*, pages 513–527. Springer-Verlag, 2001. 2

[6] A. Chorin. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics*, 57:785 – 796, 1973. 2

[7] J. Frank, G. Gottwald, and S. Reich. A hamiltonian particle-mesh method for the rotating shallow-water equations. In *Meshfree methods for partial differential equations*, volume 26 of *Lecture Notes in Computer Science Engineering*, pages 131–142, Berlin, 2003. Springer. 6

[8] J. Gee, M. Reivich, and R. Bajcsy. Elastically deforming 3D atlas to match anatomical brain images. *Journal of Computer Assisted Tomography*, 17(2):225–236, 1993. 1

[9] U. Grenander and M. Miller. Computational anatomy: An emerging discipline. *Quarterly of Applied Mathematics*, 56(4):617 – 694, 1998. 1

[10] E. Haber and J. Modersitzki. Beyond mutual information: A simple and robust alternative. In H. A. et al., editor, *Bildverarbeitung fr die Medizin*. Springer, 2005. 2

[11] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, Berlin, 2002. 6

[12] F. H. Harlow and J. E. Welsh. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8:2182–2189, 1965. 6

[13] D. D. Holm, J. E. Marsden, and T. S. Ratiu. The Euler-Poincaré equations and semidirect products, with applications to continuum theories. *Advances in Mathematics*, 137(1):1–81, 1998. 2

[14] D. D. Holm, J. T. Ratnanather, A. Trouvé, and L. Younes. Soliton dynamics in computational anatomy. *NeuroImage*, 23(Supplement 1):S170 – S178, 2004. 1

[15] S. C. Joshi and M. M. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Transactions on Image Processing*, 9(8):1357–1370, 2000. 2

[16] O. Junge, J. E. Marsden, and S. Ober-Blbaum. Discrete mechanics and optimal control. In *Proceedings of International Federation of Automatic Control Conference*, 2005. 6

[17] F. Maes, A. Collignon, D. Vandermeulun, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, 1997. 1

[18] J. Maintz and M. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, 1998. 1

[19] J. Marsden and T. Ratiu. *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. Springer, Berlin, 2 edition, 1999. 2

[20] J. E. Marsden and S. D. Ross. New methods in celestial mechanics and mission design. *Bulletin of the American Mathematical Society*, 43(1):43–73, 2005. 7

[21] S. Marsland and C. Twining. Constructing data-driven optimal representations for iterative pairwise non-rigid registration. In *Second International Workshop on Biomedical Image Registration*, number 2717 in Lecture Notes in Computer Science, pages 50–60. Springer, 2003. 3, 5

[22] S. Marsland and C. Twining. Constructing diffeomorphic representations for the groupwise analysis of non-rigid registrations of medical images. *IEEE Transactions on Medical Imaging*, 23(8):1006 – 1020, 2004. 1, 2, 4, 6

[23] M. Miller, A. Trouvé, and L. Younes. On metrics and the Euler-Lagrange equations of computational anatomy. *Annual Reviews in Biomedical Eng.*, 4:375 – 405, 2002. 1

[24] D. Mumford. *Pattern Theory and Vision*, chapter 3, pages 7 – 13. Institute Henri Poincaré, Paris, 1998. 2

[25] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The correlation ratio as a new similarity measure for multimodal image registration. *Lecture Notes in Computer Science*, 1496:1115–1124, 1998. 2

[26] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Non-rigid registration using free-form deformations: Application to breast MR images. *IEEE Transactions on Medical Imaging*, 18(8):712–721, 1999. 1

[27] P. Thompson, K. M. Hayashi, E. R. Sowell, N. Gogtay, J. N. Giedd, J. L. Rapoport, G. de Zubicara andc Andrew L. Janke, S. E. Rose, J. Semple, D. M. Doddrell, Y. Wang, T. G. van Erp, T. D. Cannon, and A. W. Toga. Mapping cortical change in alzheimers disease, brain development, and schizophrenia. *NeuroImage*, 23:S2 – S18, 2004. 1

[28] A. W. Toga. *Brain Warping*. Academic Press, 1999. 1

[29] P. Viola and W. M. Wells III. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 24(2):137–154, 1997. 1

[30] B. Zitová and J. Flusser. Image registration methods: A survey. *Image and Vision Computing*, 21:977 – 1000, 2003. 1