Insight Into Efficient Image Registration Techniques and the Demons Algorithm

Tom Vercauteren, Xavier Pennec, Ezio Malis, Aymeric Perchant, Nicholas Ayache

Insight Into Efficient Image Registration Techniques and the Demons Algorithm

Tom Vercauteren[1,2], Xavier Pennec[1], Ezio Malis[3], Aymeric Perchant[2], and Nicholas Ayache[1]

[1] Asclepios Research Group, INRIA Sophia-Antipolis, France
[2] Mauna Kea Technologies, 9 rue d'Enghien Paris, France
[3] Icare Research Group, INRIA Sophia-Antipolis, France

Abstract. As image registration becomes more and more central to many biomedical imaging applications, the efficiency of the algorithms becomes a key issue. Image registration is classically performed by optimizing a similarity criterion over a given spatial transformation space. Even if this problem is considered as almost solved for linear registration, we show in this paper that some tools that have recently been developed in the field of vision-based robot control can outperform classical solutions. The adequacy of these tools for linear image registration leads us to revisit non-linear registration and allows us to provide interesting theoretical roots to the different variants of Thirion's demons algorithm. This analysis predicts a theoretical advantage to the symmetric forces variant of the demons algorithm. We show that, on controlled experiments, this advantage is confirmed, and yields a faster convergence.

1 Introduction

As the integration of information from multiple images finds more and more applications in the fields of biomedical research and clinical applications, the efficiency of the image registration procedures becomes a crucial point for the end-users. Correspondingly there is a growing interest from the scientific community to better understand and optimize the registration procedures.

In this paper, we present an efficient approach to image registration that encompass both linear and non-linear registration with a focus on mono-modal image registration. In this setting, registration is classically performed by optimizing a similarity criterion such as the mean squared error. Literature on image registration and optimization theory already provide a wealth of algorithms that can be used to solve this problem. However they do not always use all the specificity of mono-modal image registration. *Our first contribution* is to shed a new light on this problem by showing that the tools that have recently been developed by Malis [1] in the field of vision-based robot control can be used for biomedical image registration and that they outperform the well-known optimizers. The efficient second-order minimization (ESM) technique [1] takes advantage of the specificity of mono-modal image registration to boost its convergence rate. It is

not tailored to a particular class of spatial transformations and can thus be used for a broad class of problems.

Looking at non-linear image registration, one of the most efficient methods is the demons algorithm proposed by Thirion [2]. Several variants of the algorithm have been proposed depending on how the *forces* are computed. In [3, 4] an *ad hoc* symmetrization of the demons force similar to the one proposed by Thirion was shown to improve the results of the original demons algorithm. In [5] the authors showed that the demons algorithm had connection with gradient descent schemes. However, to the best of our knowledge, the different variants of the demons have not been given a strong unified theoretical justification. *Our second and main contribution* is to show that the image registration framework we use in this work provides strong theoretical roots to the demons algorithm and that the different variants are related to the use of different optimizers. One of the main results of our theoretical analysis is to show that the symmetric forces variant is related to the ESM scheme. This study thus explains why, from a theoretical point of view, the symmetric forces demons algorithm seems to be more efficient in practice. *Our third contribution* is to provide evidence that, in practice, using symmetric forces indeed leads to a higher convergence rate.

The paper is organized as follows. In Section 2, we develop a unified framework for mono-modal image registration and show how the classical optimizer fit in the framework. Section 3 focuses on the efficient second-order minimization (ESM) with an emphasis on sound mathematical treatment. A practical example is worked out to compare the performance of the different approaches. In Section 4 we show how to extend the framework for the study of the demons algorithm. Finally Section 5 concludes the paper.

2 Registration using Newton Methods on Lie Groups

2.1 Image Registration Model

Given a *fixed image* $F(.)$ and a *moving image* $M(.)$ in a D-dimensional space, intensity-based image registration is treated as an optimization problem that aims at finding the spatial mapping that will align the fixed and moving images. The transformation $s(.)$, $\mathbb{R}^D \rightarrow \mathbb{R}^D$, $p \mapsto s(p)$, models the spatial mapping of points from the fixed image space to the moving image space. The similarity criterion $Sim(F, M \circ s)$ measures the quality of a given transformation. In this paper we will only consider the mean squared error similarity measure which forms the basis of the intensity-based image registration algorithms:

$$Sim(F, M \circ s) = \frac{1}{2} \left\| F - M \circ s \right\|^2 = \frac{1}{2 \left| \Omega_P \right|} \sum_{p \in \Omega_P} \left| F(p) - M(s(p)) \right|^2 , \quad (1)$$

where Ω_P is the region of overlap between F and $M \circ s$.

In order to register the fixed and moving images, we need to optimize (1) over a given space of spatial transformations. This can often be done by parameterizing the transformations. However most of the spatial transformations

we use do not form vector spaces but only Lie groups (e.g. rigid body, affine, projective, diffeomorphisms...), meaning that we can invert or compose these transformations and obtain a spatial transformation of the same type. We thus need to perform an optimization procedure on a Lie group such as in [1, 6, 7].

2.2 Newton Methods for Lie Groups

Optimization problems on Lie groups can often be related to constrained optimization by embedding the Lie group in an Euclidean space. The classical way of dealing with the structure of the group is to use Lagrange multipliers or when the constraints are simple to have an *ad hoc* procedure to preserve the constraints (e.g. renormalize a quaternion to have a unit quaternion). In this work we use an alternative strategy known as geometric optimization which uses local canonical coordinates [6]. This strategy intrinsically takes care of the geometric structure of the group and allows the use of unconstrained optimization routines.

Let $\mathcal{G}$ be a Lie group for the composition $\circ$. We refer the reader to the standard textbooks for a detailed treatment of Lie groups. To any Lie group can be associated a Lie algebra $\mathfrak{g}$, whose underlying vector space is the tangent space of $\mathcal{G}$ at the neutral element Id. This Lie algebra captures the local structure of $\mathcal{G}$. The Lie group and the Lie algebra are related through the group exponential which is a diffeomorphism from a neighborhood of 0 in $\mathfrak{g}$ to a neighborhood of Id in $\mathcal{G}$. Let $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$ be a basis of the Id-tangent space $T_{\mathrm{Id}}(\mathcal{G})$ corresponding to a basis of $\mathfrak{g}$. Canonical coordinates provide local coordinate charts so that for any $x \in \mathcal{G}$ in some neighborhood of s, there exists a vector $\boldsymbol{u} = \sum_i u_i \boldsymbol{e}_i \in T_{\mathrm{Id}}(\mathcal{G})$ such that $x = s \circ \exp(\boldsymbol{u}) = s \circ \exp(\sum_i u_i \boldsymbol{e}_i)$. They can be used to get the Taylor expansion of a smooth function φ on $\mathcal{G}$:

$$\varphi\left(s \circ \exp(\boldsymbol{u})\right) = \varphi(s) + J_s^{\varphi}.\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}^T.H_s^{\varphi}.\boldsymbol{u} + O(\|\boldsymbol{u}\|^3), \tag{2}$$

where $[J_s^{\varphi}]_i = \frac{\partial}{\partial u_i}\varphi(s \circ \exp(\boldsymbol{u}))\big|_{\boldsymbol{u}=0}$ and $[H_s^{\varphi}]_{ij} = \frac{\partial^2}{\partial u_i \partial u_j}\varphi(s \circ \exp(\boldsymbol{u}))\big|_{\boldsymbol{u}=0}$. It is shown in [6], that this expansion allows us to adapt the classical Newton-Raphson method by using an intrinsic update step,

$$s \leftarrow s \circ \exp(\boldsymbol{u}) \tag{3}$$

where $\boldsymbol{u}$ solves $H_s^{\varphi}.\boldsymbol{u} = -J_s^{\varphi T}.\varphi(s)$. As in the vector space case, this algorithm has a local quadratic convergence, and is independent of the chosen basis of $\mathfrak{g}$.

In many cases, using the Newton-Raphson method is not advocated or simply not possible. The Hessian matrix is indeed often difficult or impossible to compute, is not numerically well-behaved and convergence problem may arise when it is not definite positive. To address these problems in the context of non-linear least squares optimization, most of the available efficient methods (e.g. Levenberg-Marquardt) are related to the Gauss-Newton method [8].

Let $\phi(.) = \frac{1}{2}\|\boldsymbol{\varphi}(.)\|^2 = \frac{1}{2}\sum_p \varphi_p(.)^2$ be a sum of squared smooth functions. The Gauss-Newton method is based on a linear approximation of $\boldsymbol{\varphi}$ in

a neighborhood of the current estimate. From (2), we have $\boldsymbol{\varphi}(s \circ \exp(\boldsymbol{u})) = \boldsymbol{\varphi}(s) + J_s^{\boldsymbol{\varphi}}.\boldsymbol{u} + O(\|\boldsymbol{u}\|^2)$. By keeping only the linear part we obtain a quadratic approximation that we use to derive the Gauss-Newton method on a Lie group:

$$\phi(s \circ \exp(\boldsymbol{u})) = \frac{1}{2} \|\boldsymbol{\varphi}(s \circ \exp(\boldsymbol{u}))\|^2 \approx \frac{1}{2} \|\boldsymbol{\varphi}(s) + J_s^{\boldsymbol{\varphi}}.\boldsymbol{u}\|^2 . \tag{4}$$

It is well known that if $J_s^{\boldsymbol{\varphi}}$ has full rank, this equation admits a unique minimizer which is the solution of the so-called normal equations: $\left(J_s^{\boldsymbol{\varphi}T}.J_s^{\boldsymbol{\varphi}}\right).\boldsymbol{u} = -J_s^{\boldsymbol{\varphi}T}.\boldsymbol{\varphi}(s)$. By using this solution in the intrinsic update step, $s \leftarrow s \circ \exp(\boldsymbol{u})$, we get the Gauss-Newton method for Lie Groups. In a vector space, the local convergence of the Gauss-Newton (and Levenberg-Marquardt) method is in general not quadratic. In the Lie group setting, we also see that (4) is a first-order approximation. We must therefore also expect only local linear convergence.

2.3 Gauss-Newton for Image Registration

For the registration problem (1), the Gauss-Newton algorithm can be used with the following function involved in the nonlinear least squares problem:

$$\varphi_p(s \circ \exp(\boldsymbol{u})) = F(p) - M \circ s \circ \exp(\boldsymbol{u})(p). \tag{5}$$

We now need to know how to compute the Jacobian $J_s^{\varphi_p}$ of this function.

In practice, we need a computational representation. By Whitney's theorem, we know that there exists an embedding Θ, $\mathcal{G} \to \mathbb{R}^N$, $s \mapsto \Theta(s)$ of the Lie group in an Euclidean space. This embedding also allows us to represent the Lie algebra. An example is the matrix representation of the common spatial transformations (e.g. rigid body, affine, projective) in homogeneous coordinates. In practice, this Euclidean representation is used to compute the spatial transformation (using e.g. matrix multiplication in homogeneous coordinates). Let us denote $w(\Theta(s), p)$ the expression, in the Euclidean embedding space $\mathbb{R}^N$, of the transformation of a point $p \in \mathbb{R}^D$ through the mapping $s \in \mathcal{G}$. Using this Euclidean representation, the chain rule and the fact that the differential map of the exponential at Id is the identity, the Jacobian of (5) can be decomposed as (cf. appendix):

$$J_s^{\varphi_p} = \frac{\partial}{\partial \boldsymbol{u}^T} \varphi_p(s \circ \exp(\boldsymbol{u}))\Big|_{\boldsymbol{u}=0} = -\nabla_p^T (M \circ s).J^{w_p}.\boldsymbol{e}_{\Theta}, \tag{6}$$

where $\nabla_p(M \circ s)$ is the gradient of the warped moving image ($D \times 1$ vector), $J^{w_p} = \frac{\partial w(X,p)}{\partial X^T}\big|_{X=\Theta(\mathrm{Id})}$ is the derivative of the mapping action expressed the Euclidean embedding space ($D \times N$ matrix) and $\boldsymbol{e}_{\Theta} = [\Theta(\boldsymbol{e}_1), \ldots, \Theta(\boldsymbol{e}_n)]$ stacks the basis vectors of $\mathfrak{g}$ expressed in the Euclidean embedding space ($N \times n$ matrix). A practical case for 2D rigid-body registration is given in Section 2.3.

3 Efficient Second-Order Minimization (ESM)

Image registration (especially mono-modal) is not any generic optimization problem, the algorithms can take advantage of the specificity of the problem to develop more efficient schemes. We focus on the efficient second-order minimization

(ESM) procedure of [1] that uses the following fact: when the images are aligned with the optimal spatial transformation s^{opt}, the fixed image and the warped image as well as their gradient should be very close to each other.

The main idea behind the ESM is that we can use this information to improve the search direction of the Newton methods. The Newton-Raphson uses the value of φ_p, its first and second derivatives around 0 to build a second-order polynomial approximation of φ_p. The Gauss-Newton case discards the second derivative and can thus only build a first-order polynomial. What we do with the ESM is to use the value of φ_p, its first derivative around 0 as well as its first derivative around s^{opt} to build a second-order polynomial without the need of second derivative information. The ESM is thus a second-order minimization method that does not need the computation of the Hessian matrix.

3.1 A Second-Order Linearization

With the ESM, the information about the Hessian that is discarded with the Gauss-Newton iteration is recovered with a Taylor expansion of a Jacobian calculated at the optimal transformation. Such an information can only be used in the image registration settings because we should have $\nabla_p M \circ s^{\text{opt}} \approx \nabla_p F$ up to a noise term. In order to use this very special property, let us define a generalization of the Jacobian used in Section 2.2:

$$J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) = \frac{\partial}{\partial \boldsymbol{v}^T} \boldsymbol{\varphi}(s \circ \exp(\boldsymbol{v}))\Big|_{\boldsymbol{v}=\boldsymbol{u}}. \tag{7}$$

Note that $J_s^{\boldsymbol{\varphi}}(0) = J_s^{\boldsymbol{\varphi}}$. By using a first-order Taylor expansion around 0 we get:

$$J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) = J_s^{\boldsymbol{\varphi}}(0) + \boldsymbol{u}^T.H_s^{\boldsymbol{\varphi}} + O(\|\boldsymbol{u}\|^2),$$

that can be rewritten as $\boldsymbol{u}^T.H_s^{\boldsymbol{\varphi}} = J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) - J_s^{\boldsymbol{\varphi}}(0) + O(\|\boldsymbol{u}\|^2)$. By incorporating this expression into (2), this provides us with a true second-order approximation:

$$\begin{aligned} \boldsymbol{\varphi}\left(s \circ \exp(\boldsymbol{u})\right) &= \boldsymbol{\varphi}(s) + J_s^{\boldsymbol{\varphi}}(0).\boldsymbol{u} + \frac{1}{2}\left(J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) - J_s^{\boldsymbol{\varphi}}(0)\right).\boldsymbol{u} + O(\|\boldsymbol{u}\|^3) \\ &= \boldsymbol{\varphi}(s) + \frac{1}{2}\left(J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) + J_s^{\boldsymbol{\varphi}}\right).\boldsymbol{u} + O(\|\boldsymbol{u}\|^3) \end{aligned} \tag{8}$$

The non-linear least squares problem of Section 2.2 can thus be revisited to get a second-order approximation of (4):

$$\phi(s \circ \exp(\boldsymbol{u})) = \frac{1}{2}\|\boldsymbol{\varphi}(s) + \frac{1}{2}\left(J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}) + J_s^{\boldsymbol{\varphi}}\right).\boldsymbol{u}\|^2. + O(\|\boldsymbol{u}\|^3) \tag{9}$$

The computation of $J_s^{\boldsymbol{\varphi}}(\boldsymbol{u})$ is a difficult problem in the general setting. Even if we get a closed-form expression of it, a minimization problem that involves this term might not be easy to solve in practice. In order to be able to use (9), we need to use the special properties of our optimization problem.

From the current transformation s, the optimum step $\boldsymbol{u}_s^{\text{opt}}$ that an optimizer can make is such that $s^{\text{opt}} = s \circ \exp(\boldsymbol{u}_s^{\text{opt}})$. From a computational point of view,

the main result of the ESM procedure is that, for this optimal step, the product $J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}_s^{\text{opt}}).\boldsymbol{u}_s^{\text{opt}}$ is linear in $\boldsymbol{u}_s^{\text{opt}}$. This allows for a simple minimization of (9).

The idea is to replace the gradient of the optimally warped image $M \circ s^{\text{opt}} = M \circ s \circ \exp(\boldsymbol{u}_s^{\text{opt}})$ by its *equivalent*, the gradient of the fixed image F. We then get a simple linear approximation: $J_s^{\boldsymbol{\varphi}}(\boldsymbol{u}_s^{\text{opt}}).\boldsymbol{u}_s^{\text{opt}} \approx \nabla_p^T F.J^{w_p}.\boldsymbol{e}_\Theta.\boldsymbol{u}_s^{\text{opt}}$ as shown in the appendix. This can be used with (8) to get:

$$\boldsymbol{\varphi}\left(s \circ \exp(\boldsymbol{u}_s^{\text{opt}})\right) = \boldsymbol{\varphi}(s) + J_s^{\text{ESM}}.\boldsymbol{u}_s^{\text{opt}} + O(\|\boldsymbol{u}_s^{\text{opt}}\|^3) \tag{10}$$

$$J_s^{\text{ESM}_p} \triangleq -\frac{1}{2}\left(\nabla_p^T F + \nabla_p^T (M \circ s)\right).J^{w_p}.\boldsymbol{e}_\Theta \tag{11}$$

where we omit the image noise and where J^{w_p} and $\boldsymbol{e}_\Theta$ are given in Section 2.3. This efficient procedure can thus be summarized by the following algorithm:

Algorithm 1 (ESM and Gauss-Newton for Image Registration)

- *Choose a starting spatial transformation s*
- *Iterate until convergence:*
  - *Given s, let*
    - $J^p = -\frac{1}{2}\left(\nabla_p^T F + \nabla_p^T (M \circ s)\right).J^{w_p}.\boldsymbol{e}_\Theta$ *for ESM*
    - $J^p = -\nabla_p^T (M \circ s).J^{w_p}.\boldsymbol{e}_\Theta$ *for Gauss-Newton*
  - *Compute the update $\boldsymbol{u}$ by solving the linear system $(J^T.J).\boldsymbol{u} = -J^T.\boldsymbol{\varphi}(s)$ using e.g. a QR factorization of J*
  - *Let $s \leftarrow s \circ \exp(\lambda \boldsymbol{u})$, with $\lambda = 1$ or is given by a line search*

Note that the two options have the same computational complexity since $\nabla_p F$ needs only be computed once during initialization.

3.2 A Practical Example: 2D Rigid Body Transformations

Let us now focus on the optimization of (1) for the Lie group $SE(2)$ of 2D rigid body transformations. In order to use the optimization method presented in Algorithm 1, we need to know what the corresponding Lie algebra $\mathfrak{se}(2)$ is and to be able to compute the exponential map and the necessary Jacobian.

A 2D rigid body transformation r is composed of a rotation of angle α followed by a translation $\tau = (\tau^x, \tau^y)$. This Lie group $SE(2)$ can be represented using homogeneous coordinates by a 3×3 matrix group of the form $\Theta(r) = \left[\begin{smallmatrix} R_\alpha & \tau \\ 0 & 1 \end{smallmatrix}\right]$, where R_α is a rotation matrix. Since we have a matrix Lie group, the exponential map coincides with the matrix exponential. In this special case, we even have a closed-form expression of the matrix exponential. Thanks to this matrix representation (which is the Euclidean embedding space used in Section 2.3), we see that the Lie Algebra can be represented by the vector space of matrices of the form $\left[\begin{smallmatrix} dR_\alpha & d\tau \\ 0 & 0 \end{smallmatrix}\right]$, where dR_α is any skew-symmetric matrix and $d\tau$ is any vector. We thus see that a convenient basis of $\mathfrak{se}(2)$ is given (in matrix form) by $\Theta(\boldsymbol{e}_1) = \left[\begin{smallmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right]$, $\Theta(\boldsymbol{e}_2) = \left[\begin{smallmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right]$ and $\Theta(\boldsymbol{e}_3) = \left[\begin{smallmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{smallmatrix}\right]$. In order to use Algorithm 1, the only expression we still need to compute is J^{w_p}. The spatial

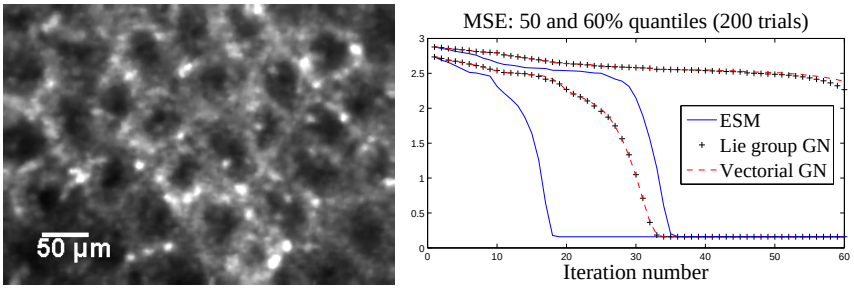


Fig. 1. Simple convergence experiment using two consecutive live mouse colon images of a dynamic fibered confocal microscopy (FCM) sequence (left, image courtesy of D. Vignjevic, S. Robine, D. Louvard, Institut Curie, Paris). We initialize the different optimizers with a random rigid body transformations to compare them. The random generator is Gaussian centered around the optimal transformation (validated by an expert), uses $\sigma_\alpha = 0.3$ rad for the rotation part and $\sigma_\tau = 10$ pixels for the translation parts so as to get a sufficient number of divergent optimizations. The ESM is both faster to converge with 50% of the trials converging in less than 19 iterations vs. 34 for the Gauss-Newton and more robust as 60% converge in less than 36 iterations with ESM but we never reach 60% of convergence with the Gauss-Newton.

transformation $r(p)$ of a point p through a 2D rigid body transformation r is a simple matrix multiplication and this leads to $J^{w_p} = \begin{bmatrix} p_x & 0 & 0 & p_y & 0 & 0 & 1 & 0 & 0 \\ 0 & p_x & 0 & 0 & p_y & 0 & 0 & 1 & 0 \end{bmatrix}$. After some basic simplifications we obtain a simple expression: $J^{w_p}.\boldsymbol{e}_\Theta = \begin{bmatrix} -p_y & 1 & 0 \\ p_x & 0 & 1 \end{bmatrix}$.

Registration results: In the context of tracking for vision-based robot control, a detailled comparison of the optimization schemes showed that, for the space of homographies, the ESM outperformed classical solutions [9]. In this section, we compare the performance of the ESM optimizer with respect to the Gauss-Newton optimizer on a real-life biomedical image registration problem. A $2D+t$ dynamic sequence is acquired with a fibered confocal microscope (FCM) and we need to perform a rigid body registration between the consecutive frames. In order to get a statistically meaningful example, we chose two representative frames and compared the optimizers with random starting points. Since the emphasis is on the comparison of the various schemes and not on the final performance, no multi-resolution scheme was used. Our results in Fig. 1 show that the analysis of [9] can be extended to the problem of biomedical image registration. We indeed see that for rigid body registration the ESM has a faster convergence rate and is more robust than the Gauss-Newton optimizer.

4 An Insight into the Demons Algorithm

In [2], the author proposed to consider non-linear registration as a diffusion process. He introduced *demons* that push according to local characteristics of

the images in a similar way Maxwell did for solving the Gibbs paradox. The forces are inspired from the optical flow equations and the method alternates between computation of the forces and their regularization by a simple Gaussian smoothing. This results into a computationally efficient algorithm. Several teams [10, 5, 11] have worked towards providing theoretical roots to the demon's in order to understand the underlying assumptions and potentially modify them.

The goal of this section is twofold. We first go one step further in providing theoretical explanations of the demons and show that the different variants of of this algorithm can all be cast into the image registration framework derived above. One of the main results of this analysis is to show that the symmetric forces demons can be cast to the ESM optimization method of [1]. This variant should therefore be the most efficient one. Our second goal is thus to verify if the theoretical advantage of the symmetric variant are noticeable in practice.

4.1 An Alternate Optimization Framework

In order to end-up with the global minimization of a well posed criterion, it was proposed in [11] to introduce a hidden variable in the registration process: correspondences. The idea is to consider the regularization criterion as a prior on the smoothness of the transformation s. Instead of requiring that point correspondences between image pixels (a vector field c) be exact realizations of the transformation, one allows some error at each image point. Considering a Gaussian noise on displacements, we end-up with the global energy:

$$E(c,s) = \|\frac{1}{\sigma_i}(F - M \circ c)\|^2 + \frac{1}{\sigma_x^2}\operatorname{dist}(s,c)^2 + \frac{1}{\sigma_T^2}\operatorname{Reg}(s) \tag{12}$$

where σ_i accounts for the noise on the image intensity, σ_x accounts for a spatial uncertainty on the correspondences and σ_T controls the amount of regularization we need. We classically have $\operatorname{dist}(s,c) = \|c - s\|$ and $\operatorname{Reg}(s) = \|\nabla s\|$ but the regularization can also be modified to handle fluid-like constraints [11].

The interest of this auxiliary variable is that an alternate optimization over c and s decouples the complex minimization into simple and very efficient steps:

Algorithm 2 (Demons Algorithm as an Alternate Optimization)

- *Choose a starting spatial transformation (a vector field) s*
- *Iterate until convergence:*
  - *Given s, compute a correspondence update field $\boldsymbol{u}$ by minimizing $E_s^{\text{corr}}(\boldsymbol{u}) = \|F - M \circ (s + \boldsymbol{u})\|^2 + \frac{\sigma_i^2}{\sigma_x^2}\|\boldsymbol{u}\|^2$ with respect to $\boldsymbol{u}$*
  - *If a fluid-like regularization is used, let $\boldsymbol{u} \leftarrow K_{\text{fluid}} \star \boldsymbol{u}$. The convolution kernel will typically be a Gaussian kernel.*
  - *Let $c \leftarrow s + \boldsymbol{u}$*
  - *If a diffusion-like regularization is used, let $s \leftarrow K_{\text{diff}} \star c$ (else let $s \leftarrow c$). The convolution kernel will also typically be a Gaussian kernel.*

We focus on the first step of this alternate minimization and refer the reader to [11] for a detailed coverage of the regularization questions.

4.2 The Symmetric Forces Demons as an ESM Optimization

As one can see in Algorithm 2, the minimization of $E_s^{\text{corr}}(\boldsymbol{u})$ is very close to the mean squared error image registration problem (1) we have been focusing on. The space of free-form deformations used within the demons algorithm is a simple vector space. It is therefore a trivial Lie group where the group composition is the addition of free-form deformation fields, and the group exponential simply maps a free-form deformation field onto itself. This implies that the optimization step $s \circ \exp(\boldsymbol{u})$ we have been using so far is simply expressed by an addition $s + \boldsymbol{u}$. The only remaining difference lies in the term $\|\boldsymbol{u}\|^2$. We now show that the same framework allows us to take care of this additional term.

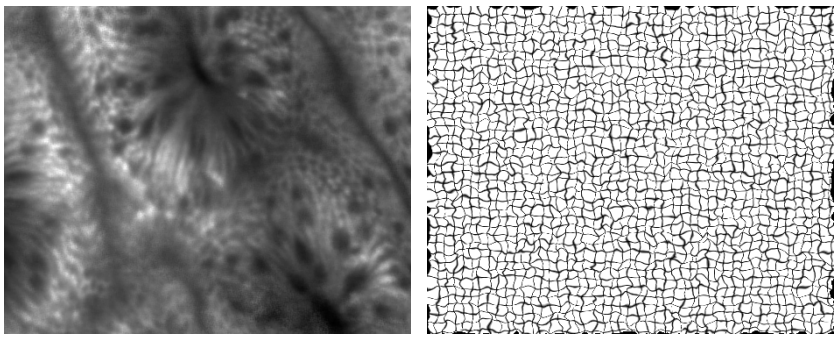
The efficient image registration tools we showed in the previous sections can easily be applied to get the following approximations:

$$F(p) - M \circ (s + \boldsymbol{u})(p) \approx F(p) - M \circ s(p) + J^p . \boldsymbol{u}(p)$$

where $J^p = -\nabla_p^T (M \circ s)$ with Gauss-Newton, $J^p = -\frac{1}{2}\left(\nabla_p^T F + \nabla_p^T (M \circ s)\right)$ with ESM and $J^p = -\nabla_p^T F$ with Thirion's rule. As shown previously, the approximation order depends on this choice of J^p. These approximations can be used to rewrite the correspondence energy used in the demons algorithm:

$$E_s^{\text{corr}}(\boldsymbol{u}) \approx \frac{1}{2\,|\Omega_P|} \sum_{p \in \Omega_P} \left\| \begin{bmatrix} F(p) - M \circ s(p) \\ 0 \end{bmatrix} + \begin{bmatrix} J^p \\ \frac{\sigma_i(p)}{\sigma_x} I \end{bmatrix} . \boldsymbol{u}(p) \right\|^2,$$

where we recall that Ω_P is the overlap between F and $M \circ s$.



(a) Original image (b) Random warped grid

Fig. 2. Experiment using a random warp on a normal human colonic mucosa image (FCM). Image Courtesy of PD. Dr. A. Meining, Klinikum rechts der Isar, Munich.

As opposed to the global transformation case (e.g. 2D rigid body transformations) we see that here, the approximations given for each pixel are independent from each other. This greatly simplifies the minimization of E_s^{corr} by splitting it

into very simple systems for each pixel. We indeed only need to solve, at each pixel p, the following normal equations:

$$\left[J^{pT} \ \frac{\sigma_i(p)}{\sigma_x} I\right] \cdot \begin{bmatrix} J^p \\ \frac{\sigma_i(p)}{\sigma_x} I \end{bmatrix} .\boldsymbol{u}(p) = -\left[J^{pT} \ \frac{\sigma_i(p)}{\sigma_x} I\right] \cdot \begin{bmatrix} F(p) - M \circ s(p) \\ 0 \end{bmatrix}$$

$$\left(J^{pT}.J^p + \frac{\sigma_i^2(p)}{\sigma_x^2} I\right) .\boldsymbol{u}(p) = -(F(p) - M \circ s(p)).J^{pT}$$

From the Sherman-Morrison formula (matrix inversion lemma) we finally have:

$$\boldsymbol{u}(p) = -\frac{F(p) - M \circ s(p)}{\|J^p\|^2 + \frac{\sigma_i^2(p)}{\sigma_x^2}} J^{pT} \tag{13}$$

We see that if we use the local estimation $\sigma_i(p) = |F(p) - M \circ c(p)|$ of the image noise, and the ESM approximation of J^p we end up with the exact expression of the symmetric forces demons algorithm. Note that σ_x then controls the maximum step length: $\|\boldsymbol{u}(p)\| \leq \sigma_x/2$.

4.3 Demons Results

To compare the performance of the different variants of the demons algorithm, we present some results using synthetic data. We use a fibered confocal microscopy image as our original image. For each random experiment, we generate a random (MRF) smooth deformation field and warp the original image. We add some random noise both to the original and the warped image. We then run the different demons algorithm starting with an identity spatial transformation.

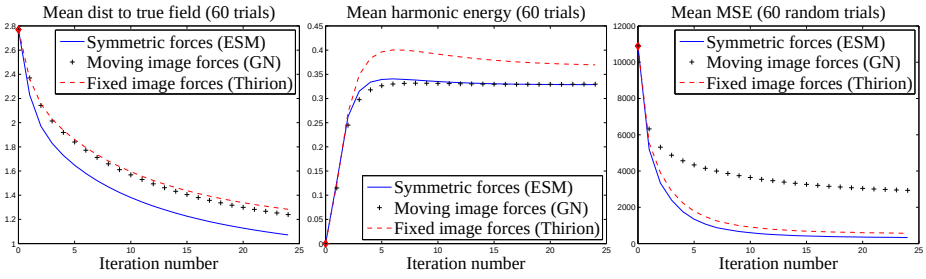


Fig. 3. Registration on 100 random experiments such as the one presented in Fig. 2. Note the faster convergence of the symmetric forces demons in terms of images intensities agreement (MSE), smoothness of the non-linear spatial transformation (harmonic energy) and more importantly in terms of distance to the actual spatial transformation.

We used the same set of parameters for all the experiments: a maximum step length of 2 pixels, a Gaussian *fluid-like* regularization with $\sigma_{\text{fluid}} = 1$ and

a Gaussian *diffusion-like* regularization with $\sigma_{\text{diff}} = 1$. As previously, no multi-resolution scheme was used because the emphasis is on the comparison of the various schemes and not the final performance. We can see on Fig. 3 that the symmetric forces variants converges faster in terms of MSE, smoothness and more importantly in terms of distance to the actual field.

5 Conclusion

We showed in this paper that some tools that have recently been developed in the field of vision-based robot control can outperform classical image registration algorithms by exploiting the special nature of the image registration problem. We have focused on mono-modal registration but the ESM scheme can also be extended to handle more complex intensity relationships. Robust estimation techniques can be used to account for outliers in the cost function and we plan to investigate on iterative intensity matching for the optimization of other simple similarity metrics such as the correlation coefficient and the correlation ratio.

The adequacy of the ESM for linear image registration led us to revisit non-linear registration and especially the demons algorithm. By using the ESM, the matrix inversion lemma and the local estimation of the image noise, we improved our understanding of the demons algorithm. This analysis predicted a theoretical advantage to the symmetric forces variant of the demons algorithm which we confirmed on the practical side.

If the confluence of the ESM theory and the alternate minimization framework of the demons algorithm indeed leads to a unified theoretical explanation of the demons, it could still be argued that all the aspects of the Lie group structure used in the ESM theory are not fully exploited there. We believe contrastingly that this Lie group point of view makes this theory much more powerfull. The final goal of understanding an algorithm is indeed to improve it. One of the main limitations of the demons algorithm is that it doesn't provide diffeomorphic transformations contrarily to the algorithms developed in [12, 13]. Our next goal will be to show how the ESM theory can be used in combination with the Lie group structure of diffeomorphic transformations to adapt the demons algorithm to this Lie group.

Appendix

Derivation of (6)**:** We apply the chain rule to $J_s^{\varphi_p} = -\frac{\partial M \circ s \circ e^{\boldsymbol{u}}}{\partial \boldsymbol{u}^T}|_{\boldsymbol{u}=0}$ and get (using the Euclidean embedding space),

$$\begin{aligned}[J_s^{\varphi_p}]_i &= -\frac{\partial M \circ s(q)}{\partial q^T}\Big|_{q=p} . \frac{\partial w(X,p)}{\partial X^T}\Big|_{X=\Theta(\mathrm{Id})} . \frac{\partial \Theta(\exp(u_i \boldsymbol{e}_i))}{\partial u_i}\Big|_{u_i=0} \\ &= -\nabla_p^T (M \circ s) . \frac{\partial w(X,p)}{\partial X^T}\Big|_{x=\Theta(\mathrm{Id})} . \Theta(\boldsymbol{e}_i),\end{aligned}$$

where we used that the differential map of the exponential at Id is the identity.

Derivation of (10)**:** We start by incorporating $\boldsymbol{u}_s^{\text{opt}}$ into $M \circ s \circ e^{\boldsymbol{v}}$ by writing it as $M \circ s \circ e^{\boldsymbol{u}_s^{\text{opt}}} \circ e^{-\boldsymbol{u}_s^{\text{opt}}} \circ e^{\boldsymbol{v}}$. By using the chain rule we find that $J_s^{\varphi_p}(\boldsymbol{u}_s^{\text{opt}})$ can be decomposed into a product of three terms. The first one is given by:

$$\frac{\partial}{\partial q^T} M \circ s \circ e^{\boldsymbol{u}_s^{\text{opt}}}(q)\Big|_{q=e^{-\boldsymbol{u}_s^{\text{opt}}} \circ e^{\boldsymbol{u}_s^{\text{opt}}}(p)} = \nabla_p^T (M \circ s^{\text{opt}}) = \nabla_p^T F + \varepsilon,$$

where ε is a noise term. The second term is the same as the one appearing in (6): $\frac{\partial w(X,p)}{\partial X^T}\big|_{X=\Theta(e^{-\boldsymbol{u}_s^{\text{opt}}} \circ e^{\boldsymbol{u}_s^{\text{opt}}})} = J^{w_p}$. And finally, the last term is given by $\frac{\partial \Theta(e^{-\boldsymbol{u}_s^{\text{opt}}} \circ e^{\boldsymbol{u}})}{\partial \boldsymbol{u}^T}\big|_{\boldsymbol{u}=\boldsymbol{u}_s^{\text{opt}}}$. This last term is in general very difficult to compute but in fact we only need to compute its product with $\boldsymbol{u}_s^{\text{opt}}$. This appears to be a directional derivative. We can thus also write it as a rate of change to get:

$$\frac{\partial \Theta(e^{-\boldsymbol{u}_s^{\text{opt}}} \circ e^{\boldsymbol{u}_s^{\text{opt}} + t\boldsymbol{u}_s^{\text{opt}}})}{\partial t}\Big|_{t=0} = \frac{\partial \Theta(e^{t\boldsymbol{u}_s^{\text{opt}}})}{\partial t}\Big|_{t=0} = \boldsymbol{e}_\Theta . \boldsymbol{u}_s^{\text{opt}}.$$

References

1. E. Malis, "Improving vision-based control using efficient second-order minimization techniques," in *Proc. IEEE ICRA'04*, Apr. 2004.
2. J.-P. Thirion, "Image matching as a diffusion process: An analogy with Maxwell's demons," *Medical Image Analysis*, vol. 2, no. 3, pp. 243–260, 1998.
3. H. Wang, L. Dong, J. O'Daniel, R. Mohan, A. S. Garden, K. K. Ang, D. A. Kuban, M. Bonnen, J. Y. Chang, and R. Cheung, "Validation of an accelerated 'demons' algorithm for deformable image registration in radiation therapy," *Physics in Medicine and Biology*, vol. 50, no. 12, 2005.
4. P. Rogelj and S. Kovačič, "Symmetric image registration," *Medical Image Analysis*, vol. 10, no. 3, pp. 484–493, Jun. 2006.
5. X. Pennec, P. Cachier, and N. Ayache, "Understanding the *Demon's Algorithm*: 3D non-rigid registration by gradient descent," in *Proc. MICCAI'99*, 1999.
6. R. Mahony and J. H. Manton, "The geometry of the Newton method on non-compact Lie-groups," *J. Global Optim.*, vol. 23, no. 3, pp. 309–327, Aug. 2002.
7. P. Y. Lee and J. B. Moore, "Gauss-Newton-on-manifold for pose estimation," *J. Indus. Management Optim.*, vol. 1, no. 4, pp. 565–587, Nov. 2005.
8. K. Madsen, H. B. Nielsen, and O. Tingleff, "Methods for non-linear least squares problems," Lecture Notes, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 1999.
9. S. Benhimane and E. Malis, "Homography-based 2d visual tracking and servoing," *Joint Issue of IJCV and IJRR on Vision and Robotics*, 2007, to appear.
10. J. Modersitzki, *Numerical Methods for Image Registration*. Oxford University Press, 2004.
11. P. Cachier, E. Bardinet, D. Dormont, X. Pennec, and N. Ayache, "Iconic feature based nonrigid registration: The PASHA algorithm," *CVIU — Special Issue on Nonrigid Registration*, vol. 89, no. 2-3, pp. 272–298, Feb.-march 2003.
12. S. Marsland and C. Twining, "Constructing diffeomorphic representations for the groupwise analysis of non-rigid registrations of medical images," *IEEE Trans. Med. Imag.*, vol. 23, no. 8, pp. 1006–1020, 2004.
13. M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes, "Computing large deformation metric mappings via geodesic flows of diffeomorphisms," *Int'l J. Comp. Vision*, vol. 61, no. 2, Feb. 2005.