

Medical image registration based on watershed transform from greyscale marker and multi-scale parameter search

Guilherme C.S. Ruppert^{a,b*}, Giovanni Chiachia^a, Felipe P.G. Bergo^c, Fernanda O. Favretto^a, Clarissa L. Yasuda^c, Anderson Rocha^a and Alexandre X. Falcão^a

^aInstitute of Computing, University of Campinas, Av. Albert Einstein 1251, Campinas, SP, Brazil; ^bCenter for Information Technology Renato Archer, Rod. Dom Pedro I km 143, Campinas, SP, Brazil; ^cNeuroimaging Laboratory, Department of Neurology, University of Campinas, R. Tessalia de Camargo 126, Campinas, SP, Brazil

(Received 17 November 2014; accepted 11 March 2015)

We propose an automatic 3D medical image registration method that combines the watershed transform from greyscale marker, to effectively reduce the number of key points needed for registration, with a robust optimisation algorithm, called multi-scale parameter search (MSPS), to quickly estimate the mapping function. We evaluate it for rigid and intra-subject registration of pre- and post-surgery MR-T1 images of the brain. The visual analysis of its effectiveness is facilitated by a colour-coding scheme. Extensive experiments show that our approach is very accurate, robust to noise and provides 3D registration in less than 40 s, with no multi-resolution image schemes needed. We also evaluate MSPS on a testbed of 12 optimisation benchmark problems, in comparison with well-known optimisers, such as particle swarm optimiser, simulated annealing and differential evolution, showing that it can also be explored in other applications.

Keywords: medical image registration; optimisation methods; biomedical image processing; watershed transform; multi-scale parameter search; global optimisers

1. Introduction

Image registration consists in aligning two or more images in a common reference system of spatial coordinates (Rueckert & Schnabel 2011). Medical image registration has been useful to combine data from the same and different imaging modalities, making it possible to visualise changes in anatomy and physiology along time and under different conditions, and to assist image-guided surgery as well as other treatments. This alignment is often done by taking one image domain as reference, finding the transformation between corresponding key points from each one of the other image domains onto the reference system, and then extending this transformation to the remaining pixels. The main problems are (i) the choice of suitable point subsets in each image domain and (ii) the determination of their mapping functions onto the reference system. We address both problems in the context of 3D rigid registration between magnetic resonance (MR) images of the human brain. In this application, the methods usually take from a few minutes (Lu et al. 2007) to several hours (Han et al. 2006) to complete registration between two images on a desktop computer.

We propose an accurate, fast, robust and automatic approach for 3D medical image registration. The method combines the watershed transform from a greyscale marker (Lotufo et al. 2002), to effectively reduce the key-point subset needed for registration, with a black-box

optimisation algorithm, called *multi-scale parameter search* (MSPS), to rapidly estimate the mapping function. The work is a considerable extension upon its conference version (Ruppert et al. 2010), presented when it was in its early stages. The present work is a considerable improvement and counts with extensive validation of both, registration function and optimisation algorithm. We evaluate it for rigid and intra-subject registration of pre- and post-surgery MR-T1 images of the brain. A comparison with the traditional sum of squared differences (SSDs) between intensities of all voxel pairs from the input images demonstrates the effectiveness of the watershed transform from the greyscale marker for key-point subset reduction. The new version of the MSPS algorithm requires less parameters and allows a variable scale size to refine the search and increase precision.

In summary, this work presents three major contributions: (i) a registration function that considerably reduces the computational time while attaining high accuracy, (ii) a new method for global optimisation of black-box problems (MSPS) and (iii) an in-depth evaluation of MSPS on a testbed of 12 optimisation benchmark problems, in comparison with other well-known optimisers, such as particle swarm optimiser (PSO), simulated annealing (SA) and differential evolution (DE), with promising results paving the way for several other applications.

*Corresponding author. Email: guilherme.ruppert@cti.gov.br

We are particularly interested in intra-subject registration of 3D MR images of the human brain. Our goal is to register pre- and post-surgical images from epilepsy patients (mostly children), who had lesioned brain tissues removed to eliminate the *foci* of the seizures. In this case, we want to evaluate the tissue differences between pre- and post-surgery by using rigid transformation (i.e. without affecting the tissue distribution in the brain) and understand their relation to cases where the patient continues to suffer seizures after surgery (Yasuda et al. 2008). Although we validate our approach for rigid registration, the method is easily extensible to global deformable registration.

Local deformable image registration plays an important role in atlas-based image segmentation, where a common strategy is to start from the result of a rigid registration process. Therefore, our method can be seen as a fast, robust and accurate approach to position the image as close as possible to the desired solution before applying local deformable registration.

The literature on image registration is vast (Audette et al. 2000; Hajnal et al. 2001; Hallpike & Hawkes 2002; Zitová & Flusser 2003; Kostelec & Periaswamy 2003; Crum et al. 2004; Zou et al. 2007; Rueckert & Aljabar 2010; Shams et al. 2010; Rueckert & Schnabel 2011; Goshtasby 2012; Sotiras et al. 2013). Maintz and Viergever > 1998) classify registration methods according to the nature of the registration basis, type of the transformation, domain of the transformation, user-interaction level, transformation search method, imaging modality and transformation subject. These seven criteria are further divided into some levels as follows. According to the nature of the registration basis, a method can be classified either as *object-based* or *image-based*. Object-based methods are those that consider image segmentation (objects, points, lines) to find the transformation (Feldmar et al. 1996; Wen 2008), while image-based methods avoid segmentation for registration (Viola & Wells 1995; Maes et al. 1997). Methods can also be classified according to the domain of the transformation either as *global* or *local*. In global approaches, only one transformation is applied to the whole image domain. When different parts of the image have distinct transformations, the method is said local. The parameters of the registration further divides the methods in those based on *parameter estimation* and *parameter search*. The former estimates the registration parameters from given point correspondences and the latter determines the parameters by finding an optimum of some criterion function defined in the parameter space (Vandermeulen et al. 1999). Other well-known terms are also used to classify registration methods: rigid vs. deformable, interactive vs. automatic, mono-modal vs. multi-modal and intra-subject vs. inter-subject (Rueckert & Schnabel 2011). There are also methods (Zhang et al. 2012) that deal with registration of tridimensional meshes, but the focus of this work is on the registration between images.

The registration method we present here is object based (the watershed transform from the greyscale marker segments the moving image into regions, reducing the set of points to their borders for registration), rigid, mono-modal and intra-subject, since we are interested in aligning MR images from the same patient; global, since the transformation for the point subsets is the same for the rest of the image domain; automatic, given that the user only provides the input images; and based on parameter search.

Many recent works on registration depend on user interaction (Pietrzyk et al. 1994; Huang et al. 2006), are only demonstrated for 2D images (Wen et al. 1996; Zou et al. 2007), provide no quantitative evaluation (Huang et al. 2006; Zou et al. 2007) or are limited to specific applications (Han et al. 2006; Lu et al. 2007). Most methods from the literature often take from a few minutes (Lu et al. 2007) to several hours (Han et al. 2006) to complete the registration of a pair of 3D MR images of the brain on a desktop computer.

Besides the image registration function, another key component of registration methods is the use of a proper optimisation algorithm. In general, a global optimisation problem consists in finding the overall optimum set of parameters for a given objective function, which is usually non-convex, containing several local minima (Gray et al. 1997). In many cases, neither analytical nor structural knowledge can be obtained from objective functions representing real-world problems. Sometimes, even weak characteristics such as monotonicity and smoothness cannot be retrieved (Auger et al. 2009). Usually, there is no other alternative to handle these problems other than the direct incorporation of its outcomes into the optimisation formulation, where the parameters being optimised set the problem internally, leading to responses from which the optimisation is guided (Hemker 2008). Such a kind of optimisation scenario is normally referred to as *black-box* optimisation and methods that cope with these problems are useful due to their generality and ease of use (Wolpert & Macready 1997; Huang et al. 2006).

To address black-box problems, many optimisation methods have been proposed in the literature with reasonable success. These methods can be roughly classified into two classes: (i) the *deterministic methods* and (ii) the *stochastic methods*. On one hand, deterministic approaches include classical *derivative-based* algorithms, in which case the derivatives are estimated by finite differences or automatic differentiation tools, and *derivative-free* algorithms, such as *Pattern Search* (PS) methods. On the other hand, stochastic methods rely on random variables to sample the search space and include bio-inspired and evolutionary algorithms (Kirkpatrick et al. 1983; Storn & Price 1997; Kennedy & Eberhart 2002).

In general, deterministic algorithms are more efficient for the minimisation of convex functions, while stochastic

algorithms, with a higher cost, better approximate the global minima of non-convex functions (Allaire 2007). The reason is that former deterministic methods tend to become locally entrapped. However, this is not the case for some recent deterministic global search methods (Floudas & Gounaris 2009; Lee 2007; Sun & Dong 2011).

In this context, here we propose a new deterministic method for global optimisation which we call MSPS. As most of stochastic and few of the deterministic approaches, this method considers the objective function as a bounded *black-box* defined over reals and, therefore, it can also be considered in functions over reals with simple bound constraints and whose structure is unknown. A simpler version of this method was previously published in da Silva et al. (2012) to optimally combine descriptors for content-based image retrieval. However, the presentation and the validation of the method in da Silva et al. (2012) were very limited. In this work, the formulation of MSPS is extended and the method is deeply validated not only in our image registration application, but also in a series of well-known benchmark functions that enable us to assess its potential as a general-purpose black-box optimiser. As we shall see, the results are promising. MSPS has provided remarkable results, both for the brain image registration problem and for benchmark problems, outperforming other state-of-the-art methods in most cases.

The tightly coupled combination of the watershed transform from the greyscale marker with the gradient-image-based criterion function, with the MSPS algorithm, makes the proposed registration approach a contribution that can possibly have a great impact in the field of medical image registration.

This paper is organised as follows. In Section 2, we present the registration function based on the watershed transform from greyscale markers. In Section 3, we introduce the MSPS optimisation method along with an illustrative example, its convergence properties and an algorithm for its implementation. Section 4 presents an evaluation of the MSPS optimiser on several optimisation benchmark problems, comparing it with other well-performing approaches in the literature. The experimental results of the registration method on clinical images are shown on Section 5. Finally, we conclude the work in Section 6 and discuss some remarks for further investigation.

2. Registration function

The registration function, often called ‘registration metric’ or ‘similarity measure’ in the literature is a function that measures the similarity between two images for a given candidate transformation. The registration function used in this work, first presented in Ruppert et al. (2010), aims at accurately and quickly accomplishing this definition, and it is described as follows.

Let $\hat{I} = (D_I, I)$ and $\hat{J} = (D_J, J)$ be two MR images for registration, where $D_I \subset \mathfrak{R}^3$ and $D_J \subset \mathfrak{R}^3$ are their image domains and the voxels $p \in D_I$ and $q \in D_J$ have intensities $I(p)$ and $J(q)$, respectively. We aim at finding the transformation M , $D_I = MD_J$, that leads the voxels in D_J to voxels in D_I (reference system) and outputs the registered image $\hat{R} = (D_I, J)$ with the intensities of \hat{J} in the same domain of \hat{I} .

Let $S_I \subset D_I$ and $S_J \subset D_J$ be subsets of these image domains, such that $S_I = MS_J$. We use a special type of watershed transform on a gradient image $\hat{G}_J = (D_J, G_J)$ of \hat{J} to extract the subset S_J as the watershed lines (borders between segmented regions). The gradient image can be interpreted as a surface (manifold) in 4D, whose basins define homogeneous regions and ridges define borders in \hat{G}_J . The watershed transform simulates a flooding on this surface with water coming from all minima. When water from different minima meet at a ridge, a barrier (watershed line) is built to avoid mix of water. The watershed lines usually represent an over segmentation of the input image. In order to considerably reduce the number of irrelevant lines to form set S_J , a superior morphological reconstruction can be applied to eliminate minima in \hat{G}_J and, therefore, the watershed transform of the reconstructed image will present considerably less watershed lines (border voxels). The watershed from the greyscale marker (Lotufo et al. 2002) computes the superior morphological reconstruction and watershed transform in a single operation, using the image foresting transform (IFT) algorithm (Falcão et al. 2004).

In the IFT framework, a gradient image is interpreted as a graph whose nodes are the voxels and arcs are defined between 26 adjacent voxels. A classical watershed transform of the gradient image would reveal too many regions (catchment basins), but we know that a superior morphological reconstruction of the gradient image from an image marker can considerably reduce its number of catchment basins. By adding a number $K > 0$ to the input image, for instance, all catchment basins, whose depth is less than K , will disappear from the watershed transform’s output, considerably reducing the number of region boundaries (watershed lines) for feature matching during registration, which makes our approach much faster than methods based on all image voxels. By specifying a path-cost function $f(\pi_p)$ applied to any path $\pi_p = \langle p_1, p_2, \dots, p \rangle$ in the image graph, the IFT algorithm minimises a cost map $V(p) = \min_{\forall \pi_p \in \Pi(p)} \{f(\pi_p)\}$, where $\Pi(p)$ is the set of all possible paths with terminus p in the graph, assigning a minimum-cost path π_p^* to every voxel p in the image. Such paths are rooted at the minima of the cost map, which turns out to be the reconstructed gradient image. The watershed from the greyscale marker algorithm also propagates a distinct label (integer number) per minimum, such that the forest of each minimum forms in a label map the labelled catchment basins. Therefore, our subset of

voxels will consist of the boundary voxels between distinct catchment basins of the reconstructed gradient image. We use 26 neighbourhood voxels (all neighbouring voxels that share at least one vertex with the central voxel) as adjacency relation for the IFT and a path-cost function

$$f(\pi_p) = \max\{G_J(p_1) + K, G_J(p_2), \dots, G_J(p)\}, \quad (1)$$

where $G_J(p)$ is the gradient magnitude computed for a voxel p , as proposed in Bergo et al. (2007), $K = 0.07G_{\max}$ for a maximum gradient value G_{\max} in the image and the factor 0.07 was determined empirically.

Next, a gradient image $\hat{G}_I = (D_I, G_I)$ of \hat{I} is also computed using the gradient method proposed in Bergo et al. (2007). When the images are registered, we expect that the most relevant borders, present in S_J , be mapped onto high gradient voxels in \hat{G}_I . Figure 1 illustrates the resulting images.

In principle, this gradient image can be created by any other method (e.g. the magnitude of the Sobel’s operator, a morphological gradient and so on). However, we have successfully used this gradient presented in Bergo et al. (2007) for many other applications (Spina et al. 2009; De miranda et al. 2010; Miranda et al. 2012). Rather than using just three directions – as in the case of a 3D Sobel’s gradient vector, for instance – the gradient image here used takes into account the intensity differences in all possible directions around a voxel within some adjacency region (e.g. a sphere of radius $r > 1$) in order to better estimate the local gradient vector.

Since our approach based on the watershed transform from the greyscale marker will output much fewer border voxels than the total number of voxels, we are considerably reducing the problem of finding the transformation M that maximises a criterion function $F(M, S_J, \hat{G}_I)$, where

$$F(M, S_J, \hat{G}_I) = \sum_{\forall q \in S_J, p = Mq \in S_I} G_I(p). \quad (2)$$

The idea here is that our registration function aims at matching region boundaries (which are closely related to the underlying object shapes – a key cue used by humans to recognise objects) by exploring the watershed lines (voxel subset) of the movable image, as boundary features, and the total reference gradient value of those lines, when they are mapped onto the reference image. Therefore, by matching the region boundaries of the movable image, as represented by its watershed lines, with the region boundaries of the reference image, as represented by voxels with higher gradient values, we expect to solve image registration.

The transformation M is then applied to the remaining voxels in D_J to compute \hat{R} . Actually, one must use $D_J = M^{-1}D_I$ and interpolate D_J to map voxel intensities for each voxel in D_I without holes in the image. Trilinear interpolation is used during all transformations to give better accuracy.

A few pre-processing steps are necessary to prepare the images for the registration. The first step is the interpolation of both images to a common voxel size using the minimum value between both images for each axis. Next, we

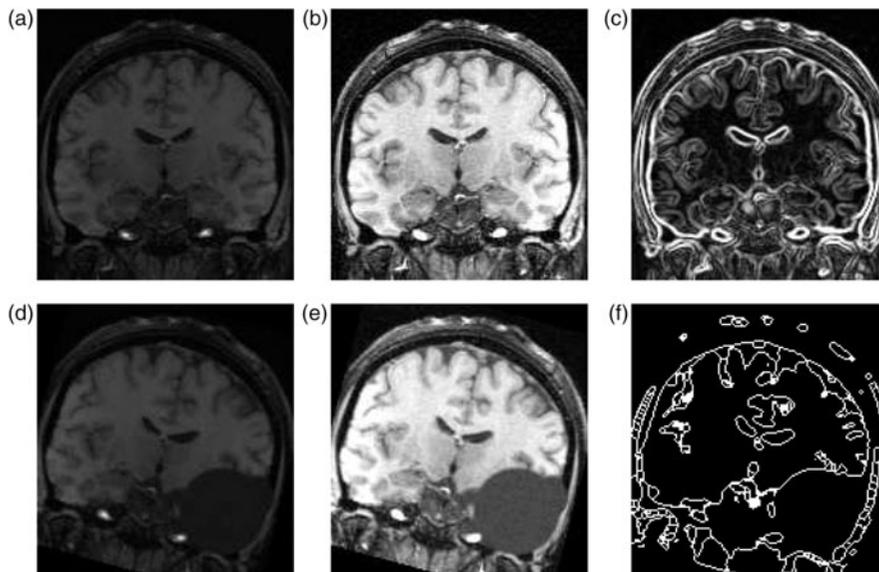


Figure 1. Slices that illustrate the pre-processing stage of a 3D registration between a MR-T1 image and another with simulated lesion: (a) Slice of an original reference image. (b) After intensity normalisation of (a). (c) Gradient image of (b). (d) Slice of the moving image. (e) After intensity normalisation of (d). (f) Watershed lines of the gradient image of (e). During registration, the accumulation of reference gradient intensities in (c) for voxels of the watershed lines in (f), which are mapped onto the reference space, is used as registration criterion for maximisation.

normalise intensities of both images by linearly mapping the intensities between 0 and the 99% mark of the accumulated histogram to the $[0, 4095]$ interval. This prevents outliers from high intensity MR artefacts, notably those from blood vessels, from forming strong borders.

For M to be a rigid transformation, we state it as a composition of a translation and three rotations. The transformation M can then be defined by six scalar parameters: $\theta = \{R_X, R_Y, R_Z, T_X, T_Y, T_Z\}$, where $\langle R_X, R_Y, R_Z \rangle$ define the rotations (angles, in degrees) and $\langle T_X, T_Y, T_Z \rangle$ define the translations (displacements, in mm). MR images provide information about the orientation of the patient so we can consider that images \hat{I} and \hat{J} are provided in the same orientation and safely start the search from $\theta = \{0, 0, 0, 0, 0, 0\}$ as initial parameters and the registration is performed by maximising F using the optimisation algorithm that is presented in Section 3.

3. Multi-scale parameter search

The registration function described in Section 2 is an example of the optimisation problem where derivatives cannot be computed, also known as a derivative-free optimisation, which relies only on the evaluation of the objective function. In this scenario, there is no alternative other than employing a heuristic to guide the search over the solution space (Conn et al. 2009).

The MSPS optimiser that we propose here uses a heuristic based on the idea that many objective function evaluations can be avoided when the search is conducted in multiple scales of displacements in the parameter space. In addition, by combining the search in different scales, the method can have a broader look at the behaviour of the objective function, which is suitable to deal with the non-convexity of the problems we intend to address, therefore, avoiding getting trapped in local minima. The method requires only a small set of parameters to be defined: the bounding constraints (maximum and minimum) of each parameter, an initialisation vector θ , the *number of scales*, the *increasing degree* of the scales and a *sampling-range decreasing factor*. The last three are described next in this section.

Our main objective is to avoid the estimation of the gradient vector of the criterion function locally, because the algorithm may be trapped into local optima, as it happens in a gradient-descent search. We could have tried to approximate that gradient vector in several scales, but it would be very unlikely to find the best direction for several scales in a space with several parameters. Therefore, we decided to explore the fact that the longest projection along some axis might indicate the best direction to go. In this way, we evaluate only those possible projections (displacements) along each axis and in several scales, and we also combine the best displacements of each axis and scale, described as follows.

Let θ be the initialisation vector containing the initial solution for the problem to be optimised, considering $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ where θ_i are the problem parameters, i.e. its dimensions. The MSPS idea is to search for displacement vectors Δ^* so that θ is iteratively updated into new positions $\theta = \theta + \Delta^*$ until an optimality criterion is met or a fixed number of iterations is performed. In order to make more reasonable moves, the method conducts the search in multiple scales, perturbing each parameter $i = \{1, 2, \dots, n\}$ in $j = \{1, 2, \dots, m\}$ displacement scales. At each iteration, MSPS evaluates $F(\theta + \Delta)$ for the displacements Δ resulting from the perturbations of each parameter in all scales and also derived from the composition of the best perturbations within the scales and among them. The displacements Δ are defined by a function that indicates how the displacements will grow across scales. This function could vary to better adapt to each application ranging from simple functions such as fixed and equidistant displacements to more complex functions. However, in this work we propose and validate a function that models the displacements as an exponential function. Therefore, depending on the exponent value, it is possible to concentrate the search closer to the current position having more density of scales in this area and yet, keep a few scales far from this point, giving the ability to circumvent local minima and also to converge faster.

An optimisation problem can be either for maximisation or minimisation of the objective function. However, for the sake of simplicity, the definitions used to explain the method are assuming a minimisation problem, but the changes necessary to address a maximisation problem are trivial.

More formally, let $\Delta_{i,j}$ be a positive displacement along the parameter axis i for scale j according to the function

$$\Delta_{i,j} = \frac{j^d}{2m^d} (u_i - l_i), \quad (3)$$

where $\mathbf{l} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^n$ are the lower and upper bounding vectors, with $u_i - l_i > 0$ for all $i = \{1, 2, \dots, n\}$, and d is the power to which the scales $j = \{1, 2, \dots, m\}$ are raised, determining how fast the displacements grow across the scales.

The method takes into account the following displacements:

- The best perturbation (if any) along each parameter axis i and scale j is

$$\Delta_{i,j}^* = (0, \dots, \Delta_{i,j}^*, \dots, 0),$$

where $\Delta_{i,j}^* \in \{\Delta_{i,j}, 0, -\Delta_{i,j}\}$ such that, for a given axis and scale, among the current position (displacement

zero) and all possible negative and positive displacements, the method selects the displacement that leads to a better criterion value, i.e.

$$F(\theta + \Delta_{i,j}^*) = \min \left\{ \begin{array}{l} F(\theta + \Delta_{i,j}), \\ F(\theta), \\ F(\theta - \Delta_{i,j}) \end{array} \right\}. \quad (4)$$

- The resulting vectors $\Delta \mathbf{s}_j = \sum_{i=1}^n \Delta_{i,j}^*$, for $j = \{1, 2, \dots, m\}$.
- The resulting vector $\Delta \mathbf{p} = \sum_{i=1}^n \Delta \mathbf{p}_i$, where

$$\Delta \mathbf{p}_i = (0, \dots, \Delta \mathbf{p}_i, \dots, 0),$$

refers to the best displacement along the scales of the parameter axis i

$$F(\theta + \Delta \mathbf{p}_i) = \min_{j=1,2,\dots,m} \left\{ F(\theta + \Delta_{i,j}^*) \right\}. \quad (5)$$

Note that $\Delta \mathbf{p}_i$ takes into account the individual values of $\Delta_{i,j}^*$ and is actually presented here for the sake of a cleaner formulation.

With all displacements evaluated, the choice of Δ^* is made so that

$$F(\theta + \Delta^*) = \min \left\{ \begin{array}{l} F(\theta + \Delta \mathbf{p}). \\ F(\theta + \Delta \mathbf{p}_i) \text{ for } i = 1, 2, \dots, n. \\ F(\theta + \Delta \mathbf{s}_j) \text{ for } j = 1, 2, \dots, m. \end{array} \right\}. \quad (6)$$

If $\theta + \Delta^*$ represents an improvement over θ , i.e. $F(\theta + \Delta^*) < F(\theta)$, then $\theta = \theta + \Delta^*$ is taken for the next iteration. Otherwise,

$$\Delta_{i,j} = \frac{\Delta_{i,j}}{2^\alpha}, \quad (7)$$

is considered in order to refine the search. The logic of this refinement is to keep exploring unvisited or relatively unexplored search space regions in early iterations and, from a certain iteration, to fine tune the best solution found so far. In MSPS, this trade-off is controlled by $\alpha > 0$.

This refinement can be seen as an increasing on the *exploitation* ability in detriment of the *exploration* capability. In earlier iterations, the aim was to keep exploring unvisited or relatively unexplored search space regions. From a certain iteration, however, the displacements must cope with the fine tuning of the solution so far. In MSPS, this tradeoff is controlled by $\alpha > 0$.

Figure 2 illustrates an iteration of the MSPS method. This toy example considers the optimisation of just two parameters that minimises some objective function using three scales. The rectangles represent the scales, and the axes represent the parameters. The origin (θ) is the solution given by the previous iteration or, in the case of the first iteration, given by a pre-defined initial position. The figure also shows other 16 labelled points. These points are the candidate points to be the solution of this iteration. The MSPS iteration basically evaluates the objective function for all these candidate points, and the one that minimises the objective function is considered the solution of the iteration. We can divide these points into three categories:

- (1) Points along the axes, which are obtained by displacing only one parameter at a time. The displacements in both positive and negative directions are given by $\Delta_{i,j}$ (Equation 3). In this example, we can see 12 points along the axis besides the origin. For each scale and parameter, the best displacement ($\Delta_{i,j}^*$) is shown labelled with a bold circle. Here, by ‘best displacement’, we mean the displacement where the function evaluation resulted in the lowest value.
- (2) Points represented with a crossed diamond are the intra-scale combination of the best displacements found in the Category (1). These points are the combination of the best displacements ($\Delta_{i,j}^*$) for each parameter within each scale. Since we have three scales in this example, this results in three points $\Delta \mathbf{s}_j$.
- (3) One last point is labelled as a star-circle diamond which is the inter-scale combination of the best displacements found in Category (1). This point ($\Delta \mathbf{p}$) combines the best displacement for each parameter among all scales.

If all these 16 evaluated points are worse than the origin, then the next iteration will have the same origin, but with smaller displacements given by Equation (7), therefore, refining the solution.

The idea behind having Categories (2) and (3) of possible candidate solutions is that, by composing the best parameters along the axes, the optimiser can walk towards the solution faster by estimating a better direction of decrease of the function. And since we are computing these displacements in different scales, we are also reducing the chances of getting trapped in local minima.

3.1 MSPS algorithm

The MSPS implementation is straightforward. In the example below, we take the advantage of composing $\Delta \mathbf{s}_j$ and $\Delta \mathbf{p}_i$ while $\Delta_{i,j}^*$ is calculated. This leads to a few control structures and requires an insignificant amount of memory. The notations through the pseudo-code are the same as in the aforementioned definitions.

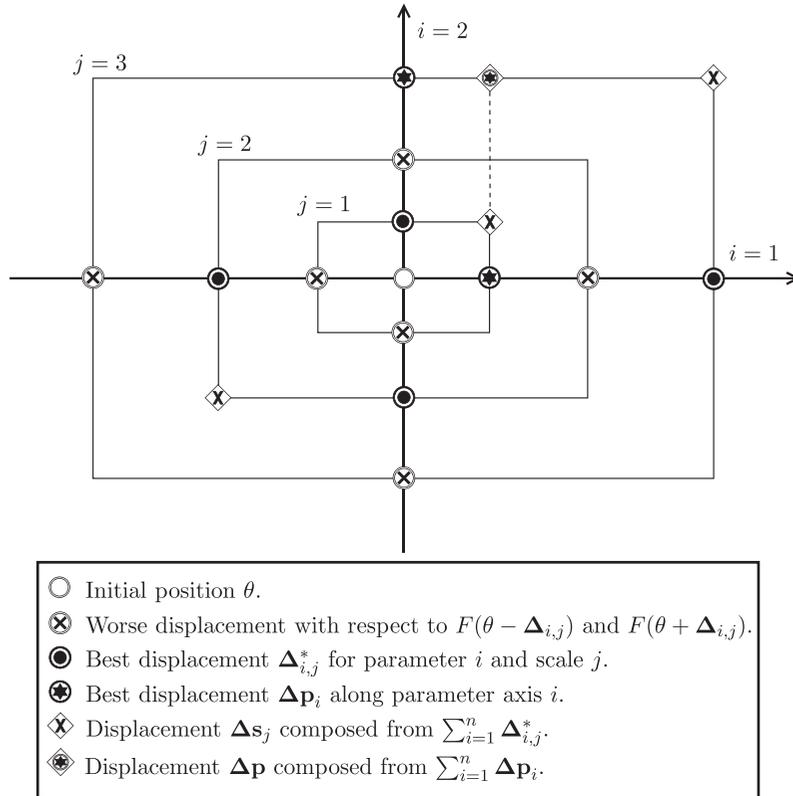


Figure 2. A toy example of an MSPS iteration. In this 2D problem, the rectangles represent the scales, and the axes the parameters. In this example, for all 16 labelled points, the objective function is evaluated and the position where the minimum value is found gives the solution of the iteration, which will be the initial position (θ) for the next iteration. The points along the axes are defined as the displacements given by $\theta \pm \Delta_{i,j}$. The points labelled with a crossed diamond ($\Delta \mathbf{s}_j$) are the combination of the best displacements for each parameter within each scale. Finally, the starry diamond ($\Delta \mathbf{p}$) is the point defined by the combination of the best displacement for each parameter among all scales. If all these 16 evaluated points are worse than the origin, then the next iteration will have the same origin, but with smaller displacements given by Equation (7), therefore, refining the solution.

Recall that the termination criterion may be the achievement of a desirable value of F , a maximum number of iterations or even $V^* = V_0$ after a number of consecutive refinements in $\Delta_{i,j}$ are considered.

4. MSPS evaluation

Although this work uses the MSPS algorithm applied to image registration, the MSPS is a general-purpose optimiser and can be applied to any black-box optimisation problem. Therefore, before we analyse the performance of this method in the image registration domain, we validated the method first on a testbed of problems for optimisation benchmarks from the literature in comparison with other state-of-the-art optimisation methods. By doing so, we gain confidence both on the algorithm effectiveness for later use in image registration and its broad range of possible applications.

4.1 Benchmark problems

In order to compare MSPS with other well-performing optimisation techniques, we evaluate the methods in a

testbed of 12 well-known benchmark problems (Pedersen 2010). These problems can be extended to a high number of dimensions and vary from convex (e.g. Sphere) to non-convex (e.g. Rastrigin) functions, and from separable (e.g. Sphere) to non-separable (e.g. Rosenbrock) problems. The QuarticNoise problem contains random noise, the Step problem is discontinuous and the problems Penalized1 and Penalized2 have constraints in the form of penalty functions. Such functions represent a very difficult class of problems for many optimisation algorithms (Xin yao et al. 1999). On evaluating them, the aim is not to show whether MSPS is better or worse than the other methods, but to find out when and why MSPS is better (or worse). After all, no single search algorithm is the best on average for all problems (Wolpert & Macready 1997).

Table 1 shows the formulations of the considered benchmark problems. All of these problems have an optimal fitness value equal to zero, although the QuarticNoise problem has added noise.

INPUT: Number n of parameters,
 Vectors \mathbf{l} and \mathbf{u} with the corresponding lower and upper bounds of each parameter,
 The number m of scales and their increasing degree d ,
 The decreasing factor α of the sampling-range,
 Vector θ with the initial parameters,
 Objective function F to be minimized.

OUTPUT: Vector θ^* with optimum parameters,
 The value V^* of the objective function F evaluated at θ^*

AUXILIARY: Search matrix $\Delta_{i,j}$ as formulated in Equation 3,
 Displacement vectors Δ , Δ^* , $\Delta\mathbf{s}$, and $\Delta\mathbf{p}$,
 Displacement vectors $\Delta\mathbf{p}_i$,
 Variable V_0 with the value of F at the beginning of the iteration,
 Variables V , V^+ , and V^- to store temporary values of F ,
 Vector Vp_i to store the best value of F along the scales of parameter i .

1. For all (i, j) , compute $\Delta_{i,j}$ by Equation 3.
2. $V^* \leftarrow F(\theta)$ and $\theta^* \leftarrow \theta$.
3. Until the termination criterion is met, do
 4. $V_0 \leftarrow V^*$, $\theta \leftarrow \theta^*$, $\Delta\mathbf{p}_i \leftarrow (0, \dots, 0)$ and $Vp_i \leftarrow V^*$ for all i .
 5. For each $j \leftarrow 1, 2, \dots, m$ do
 6. $\Delta\mathbf{s} \leftarrow (0, \dots, 0)$.
 7. For each $i \leftarrow 1, 2, \dots, n$ do
 8. $\Delta \leftarrow (0, \dots, \Delta_{i,j}, \dots, 0)$, $V \leftarrow V_0$, and $\Delta^* \leftarrow (0, \dots, 0)$.
 9. $V^+ \leftarrow F(\theta + \Delta)$ and $V^- \leftarrow F(\theta - \Delta)$.
 10. If $V^+ < V$ then $V \leftarrow V^+$ and $\Delta^* \leftarrow \Delta$.
 11. If $V^- < V$ then $V \leftarrow V^-$ and $\Delta^* \leftarrow -\Delta$.
 12. If $V < Vp_i$ then $\Delta\mathbf{p}_i \leftarrow \Delta^*$ and $Vp_i \leftarrow V$.
 13. If $V < V^*$ then $\theta^* \leftarrow \theta + \Delta^*$ and $V^* \leftarrow V$.
 14. $\Delta\mathbf{s} \leftarrow \Delta\mathbf{s} + \Delta^*$.
 15. $V \leftarrow F(\theta + \Delta\mathbf{s})$.
 16. If $V < V^*$ then $V^* \leftarrow V$ and $\theta^* \leftarrow \theta + \Delta\mathbf{s}$.
 17. $\Delta\mathbf{p} \leftarrow (0, \dots, 0)$.
 18. For each $i = 1, 2, \dots, n$ do $\Delta\mathbf{p} \leftarrow \Delta\mathbf{p} + \Delta\mathbf{p}_i$.
 19. $V \leftarrow F(\theta + \Delta\mathbf{p})$.
 20. If $V < V^*$ then $V^* \leftarrow V$ and $\theta^* \leftarrow \theta + \Delta\mathbf{p}$.
 21. If $V^* = V_0$ then $\Delta_{i,j} = \Delta_{i,j}/2^\alpha$ for all (i, j) .

Algorithm 1. MSPS.

4.2 Experimental issues

The optimisation methods compared with MSPS in this work are as follows: PSO, DE, SA and a simple variant of PS. The first three were considered due to their popularity and proven effectiveness in solving a wide range of problems (especially PSO and DE), while the PS variant was regarded because of the resemblance that MSPS has to such family of algorithms. In this last case, the aim is to stress the importance of the MSPS multi-scale property. The implementations of these four techniques were from Pedersen (2008).

Ana's important matter while evaluating responses from optimisations is how they are obtained. Normally, some index of performance on one or more problems is evaluated. Birattari & Dorigo (Birattari & Dorigo 2007) observe that reporting the central tendency of the executions of an optimisation algorithm is much more important than its best result, because the latter may be just an over-optimistic measure of performance. This is of particular importance for stochastic methods, where randomness is employed within the search. In MSPS, we have no random factors, although the solution from where our method starts the search may be random. This way, in

accordance to other originally proposed global optimisation heuristics (Storn & Price 1997; Rashedi et al. 2009), the comparisons are done in terms of the mean fitness value from a number of independent runs.

With respect to the solutions from where the methods start the search, for the benchmark problems, they were uniformly generated at random within the search intervals (Table 1) for all the optimisation methods.

For both the benchmark and the image registration problems, we present the results in tables and in performance curves. In the tables, the values are related to the mean of the best found solutions at the end of the allowed number of iterations, while in performance curves, one can obtain relative information about the progress of the optimisation over the iterations, i.e. a trade-off between the time to find a solution and its quality (Rardin & Uzsoy 2001). Each point in the curves is also an average from a number of runs.

4.3 Meta-optimisation

As most of the black-box optimisation techniques, MSPS has control parameters that need to be tuned for in order to

Table 1. Benchmark problems employed in this work and the considered search intervals (Xin yao et al. 1999).

Function	Equation	Search interval
Ackley	$f(\mathbf{x}) = e + 20 - 20 \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)\right)$	$[-30, 30]^n$
Griewank	$f(\mathbf{x}) = 1 + \frac{1}{4000}\sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right)$ $f(\mathbf{x}) = \frac{\pi}{n}(10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2(1 + 10 \sin^2(\pi y_{i+1})) + (y_n - 1)^2) + \sum_{i=1}^n u(x_i, 10, 100, 4)$	$[-600, 600]^n$
Penalized1	$y_i = 1 + (x_i + 1)/4$ $u(x_i, a, k, m) = \begin{cases} k(-x_i - a)^m, & x_i < -a \\ 0, & -a \leq x_i \leq a \\ k(x_i - a)^m, & x_i > a \end{cases}$ $f(\mathbf{x}) = 0.1(\sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2(1 + \sin^2(3\pi x_{i+1})) + (x_n - 1)^2(1 + \sin^2(2\pi x_n))) + \sum_{i=1}^n u(x_i, 5, 100, 4)$	$[-50, 50]^n$
Penalized2	with $u(\cdot)$ from above	$[-50, 50]^n$
QuarticNoise	$f(\mathbf{x}) = \sum_{i=1}^n (ix_i^4 + \text{random}[0, 1])$	$[-1.28, 1.28]^n$
Rastrigin	$f(\mathbf{x}) = \sum_{i=1}^n (x_i^2 + 10 - 10 \cos(2\pi x_i))$	$[-5.12, 5.12]^n$
Rosenbrock	$f(\mathbf{x}) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$[-100, 100]^n$
Schwefel1-2	$f(\mathbf{x}) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j\right)^2$	$[-100, 100]^n$
Schwefel2-21	$f(\mathbf{x}) = \max\{ x_i : i \in \{1, \dots, n\}\}$	$[-100, 100]^n$
Schwefel2-22	$f(\mathbf{x}) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$[-10, 10]^n$
Sphere	$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$	$[-100, 100]^n$
Step	$f(\mathbf{x}) = \sum_{i=1}^n (\lfloor x_i + 0.5 \rfloor)^2$	$[-100, 100]^n$

achieve its best performance. In our case, these parameters are the number of scales m , its increasing degree d and the sampling-range decreasing factor α . The tuning of these parameters can be done through manual experimentation. In this case, it is not clear whether a given set of parameter values really reflects the best choice one would have. One way to find optimal values for the control parameters is to conduct an exhaustive search on their combinations. However, both manual experimentation and grid-based search are expensive to execute. While human effort is costly, the exhaustive search for good combinations of parameters demands an exponentially increasing computational time with the number of parameters.

Another way to find optimum values for the control parameters is to conceive the optimisation as a problem to be optimised on its own, i.e. to employ an additional layer of optimisation. This concept is called *Meta-Optimization* herein and, in our case, is a key issue for the experiments to be fair. The idea may also be found in the literature under other names, such as *Meta-Evolution*, and has shown to be

comparatively cheap to execute and yet reaching better performance (Smit & Eiben 2009; Pedersen 2010).

Similar to most optimisation problems, the meta-optimisation also has boundaries within which the control parameters are searched. These intervals are specific for each optimisation method and are presented in Table 2.

The *Local Uniform Sampling* (LUS) technique was chosen to be the meta-optimiser for all evaluated optimisation approaches and problems because it can yield good results in a few iterations, which is of great importance in such a scenario. This method is specially designed to quickly optimise simpler problems and is often able to achieve good results on complex problems as well (Pedersen 2010).

The meta-optimisation process can take into account a single optimisation problem at a time or a group of related problems simultaneously. The measure which guides the meta-optimiser can be called *meta-fitness* and, in this paper, depending on the type of the problem, the meta-fitness value can be either the summation of the fitness values from

Table 2. Boundaries for the control parameters search space. For MSPS, the parameters are as in Section 3. With respect to PSO and DE parameters, their meaning is of common knowledge. The SA method employed takes r as the sampling-range factor, α and β as the starting and ending value of the movement-probability weight, and T as the number of iterations between resets (Pedersen 2008).

Method	Boundaries
MSPS	$m \in [1, 40]$, $d \in [0, 7]$, $\alpha \in (0, 2]$
PSO	$S \in \{1, \dots, 200\}$, $\omega \in [-2, 2]$, $\phi_p \in [-4, 4]$, $\phi_g \in [-4, 4]$
DE	$NP \in \{3, \dots, 200\}$, $CR \in [0, 1]$, $F \in [0, 2]$
SA	$r \in [1e - 5, 1]$, $\alpha \in [1e - 5, 1]$, $\beta \in [1e - 5, 1]$, $T \in [15, \#iter.]$
PS	nonparametric, as in (Pedersen 2008)

several runs on a single problem or the summation of the fitness values of a single run on several problems.

4.4 Results

Table 3 shows the MSPS optimal parameter set-ups for each optimisation problem obtained from the meta-optimisation. One can see that for many problems, the best choice was to optimise them considering just one scale in the search. This kind of behaviour is not necessarily surprising and we present a more detailed analysis with the performance results. Once we have determined these control parameters for all methods, the benchmark problems were one more time optimised with such parameters so that we could obtain the performance curves on every problem. As in the meta-optimisation, we carried out 50 runs of each method with the same number of iterations. Each point of the curves presented in Figures 3 and 4 expresses the mean fitness value from these runs at each iteration.

Two general aspects can be drawn from the results. The first one is that MSPS outperformed the other methods in 6 out of the 12 problems. In some cases, the advantage was remarkable (e.g. Ackley). The second aspect is that MSPS has always provided acceptable solutions in comparison with the other methods. Except for the QuarticNoise and the

Table 3. Optimal set-ups of the control parameters obtained from the meta-optimisation of MSPS.

Problem	Scales	Degree	α
Ackley	1	n.a.	1.58
Griewank	1	n.a.	1.52
Penalized1	2	3.67	1.31
Penalized2	2	2.59	1.55
QuarticNoise	12	0.61	1.89
Rastrigin	10	4.58	1.66
Rosenbrock	1	n.a.	0.82
Schwefel1-2	1	n.a.	1.01
Schwefel2-21	1	n.a.	1.29
Schwefel2-22	1	n.a.	1.58
Sphere	1	n.a.	1.58
Step	10	2.97	1.42

Note: The parameters are as described in Section 3.

Schwefel1-2 optimisations, the results of our method were always within the best two.

It can be observed in Figure 4 that there is a sudden drop in the Step function plot for the MSPS method. This is due to the fact that the Step function (a convex piecewise function) suddenly drops to the optimum value equal to zero whenever all parameters lie in the interval $(-0.5, 0.5)$. This plot represents the mean fitness value (throughout the iterations) for 50 randomly initialised runs of MSPS using the Step function. The reason for the sudden drop to a mean fitness value equal to zero at a given iteration in the interval $[1500, 2000]$ is that only one optimisation run (out of 50) had not already reached the minimum before that iteration. The drop occurred after this remaining run evaluates a point in the search space whose all parameters were in the interval $(-0.5, 0.5)$. Note that MSPS was the only method able to find optimum values for the Step function in all 50 runs, and we accredit this to the systematic search carried out by MSPS in multiple scales.

From the 12 problems, 5 are non-convex, namely Ackley, Griewank, Penalized1, Penalized2 and Rastrigin. For these, the number of local minima increases exponentially as the dimension of the function increases. The results are then much more important since they reflect the method's ability to escape from local optima and to locate near-global optimum solutions. When analysing the performance of MSPS in such problems, we can see that it outperformed the other methods in all cases. In addition, the plateaus in the performance curves suggest that, in many occasions, some methods become trapped in local minima. This is specially the case of PS, which is the MSPS deterministic contestant. With no doubt, these results show the MSPS's ability to handle non-convexity even when the search considers only one scale.

The convex problems are Rosenbrock, Schwefel1-2, Schwefel2-21, Schwefel2-22, Sphere and Step. With exception of Step, all others performed better with just one scale, which makes sense considering that the search movements always follow the same direction until the global minimum region is reached. In the case of Step, the multiple scales were fundamental as they allowed the method to overcome the discontinuities of the problem. Different from PS, which became trapped at a certain

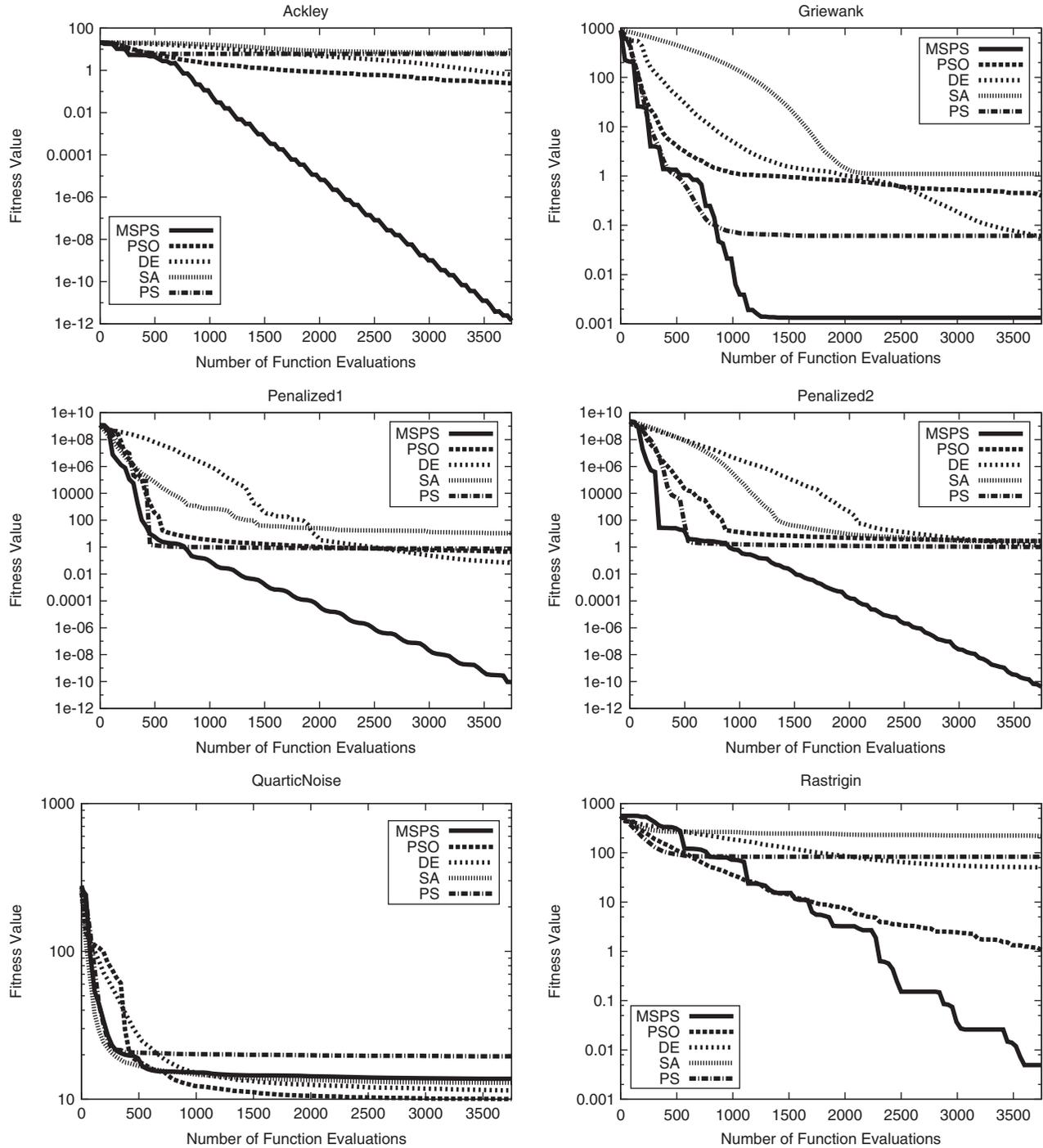


Figure 3. Comparison of the optimisation progress among the employed techniques. The control parameters of MSPS are as shown in Table 3. For all methods, the parameters were meta-optimised by considering one benchmark problem at a time for up to 3750 objective function evaluations. Each observation in the plots is a mean fitness value for 50 optimisation runs and the optimum solution for all problems has value zero.

moment, the multiscale property of MSPS enabled it to keep searching until the global optimum was found.

It is possible to observe that whenever PS had good performance, MSPS also performed well. This is the case with the optimisation of the simple Sphere and the

Schwefel2-22 problems. As pointed out before, deterministic approaches are good at optimising convex problems. The MSPS's performance confirms this idea. For Rosenbrock, its tricky and rapidly increasing shape made PSO to approach the minimum faster than MSPS. In the

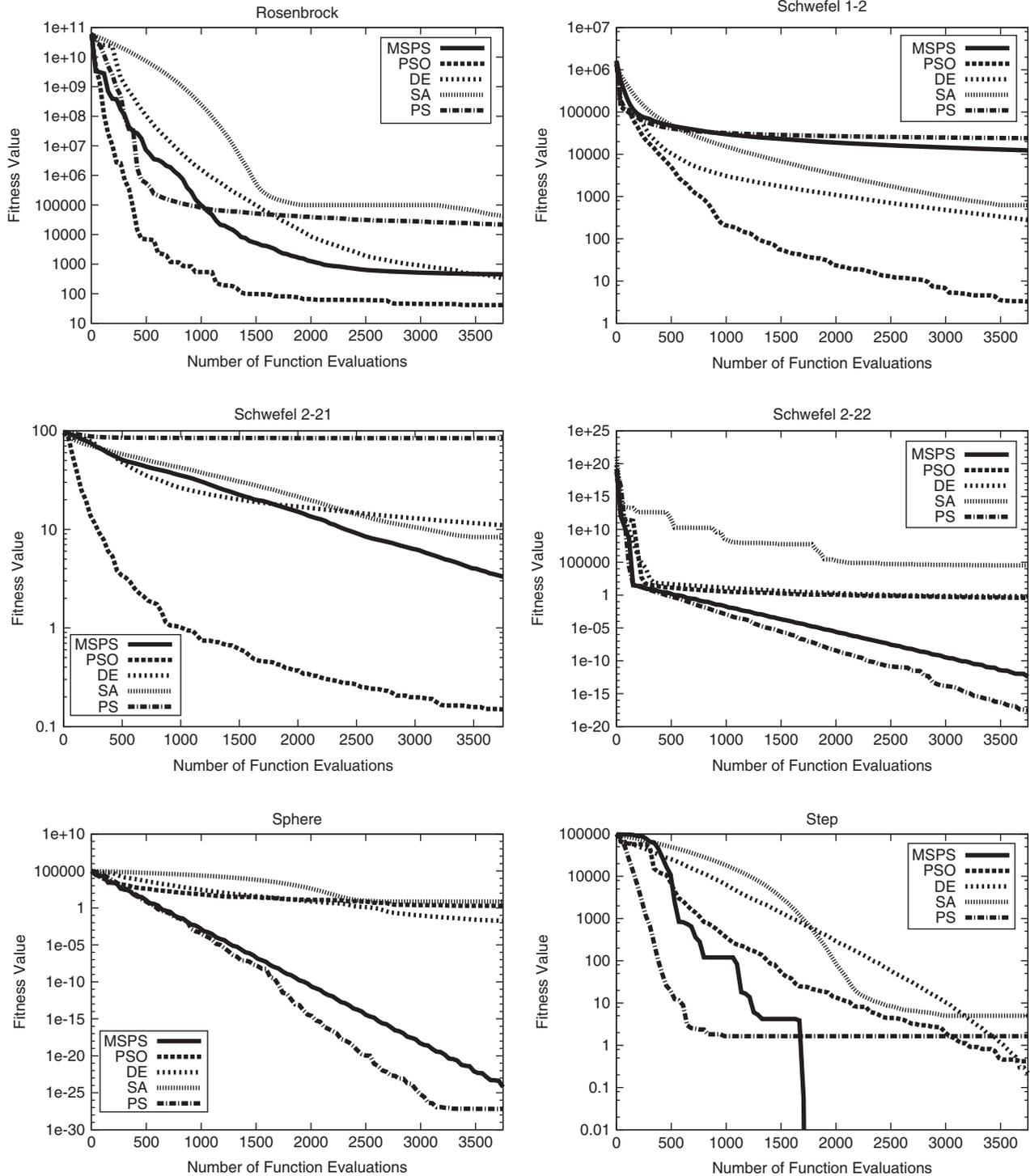


Figure 4. Comparison of the optimisation progress among the employed techniques. The control parameters of MSPS are as shown in Table 3. For all methods, the parameters were meta-optimised by considering one benchmark problem at a time for up to 3750 objective function evaluations. Each observation in the plots is a mean fitness value for 50 optimisation runs and the optimum solution for all problems has value zero.

case of Schwefel2-21, the result is acceptable, and for Schwefel1-2, MSPS did not perform well. At last, for the QuarticNoise problem, it seems that the noisy responses did not allow MSPS to fine tune the search.

In Table 4, we can see the mean fitness values at the end of the same optimisations from the plots. One observation that can be derived from these numbers is that, differently from MSPS, PS and even DE, PSO was weak in

Table 4. Mean fitness value achieved by each optimisation method for 50 optimisation runs.

Problem	MSPS	PSO	DE	SA	PS
Ackley	1.22e - 12	0.25	0.62	6.82	5.99
Griewank	1.33e - 3	0.40	5.55e - 2	1.10	6.10e - 2
Penalized1	4.70e - 11	0.46	6.76e - 2	10.81	0.77
Penalized2	3.95e - 11	2.88	1.29	2.88	1.01
QuarticNoise	13.75	10.12	11.56	13.04	19.70
Rastrigin	4.11e - 3	1.07	50.53	219.81	83.28
Rosenbrock	453.60	41.36	327.94	4.30e + 4	2.20e + 4
Schwefel1-2	12.38e + 3	3.23	280.55	623.68	2.40e + 4
Schwefel2-21	3.31	0.15	11.01	8.36	84.23
Schwefel2-22	3.69e - 13	0.38	0.66	3.38e + 4	2.95e - 18
Sphere	5.93e - 25	1.98	1.87e - 2	7.35	6.52e - 28
Step	0.00	0.42	0.22	5.00	1.64

Note: The best performances are highlighted in bold.

fine tuning. Although PSO performed best in four problems, it seems that it has some difficulties for improving near global optimum solutions.

From this point on, we have a good idea about the MSPS capabilities in dealing with different types of problems. It is important to emphasise that our analysis is restricted to the given scenario, where the number of allowed function evaluations is small. One more time, we would like to remark that there is no technique whose performance is absolutely better (Wolpert & Macready 1997). However, it is possible to have some relative idea from the results so far: MSPS performed better in six problems, PSO in four and PS in two, while DE had a reasonable performance and SA performed poorly. Apparently, MSPS has great potential in solving non-convex problems while preserving capabilities common to local search, such as fine tuning near optimum solutions. This is exactly the kind of behaviour we expect in order to solve real-world problems such as image registration.

5. Registration experiments and results

The images used in this work are volumetric T1-weighted MR images with image resolutions of either $0.98 \times 0.98 \times 1.0$ mm or $0.98 \times 0.98 \times 1.5$ mm. MR intensities range from 0 to 4095 and image sizes range from $256 \times 256 \times 100$ to $256 \times 256 \times 160$. The parameters of the MSPS algorithm were obtained by meta-optimisation using the LUS optimiser. The best parameters found were $m = 3$ (number of scales), $d = 1.106$ (increasing degree) and $\alpha = 1.151$ (decreasing factor).

We evaluated the registration accuracy of the proposed registration method in four sets of experiments. In the first set, we used MR-T1 images from control subjects transformed by known rigid transforms. In the second set, we used the same transformed images from set 1, but added synthetic (phantom) lesions to the image being registered. In the third set, we added noise to the control

images to measure robustness to noise. Finally, in the fourth set, we used pairs of clinical MR-T1 images from patients who underwent brain surgery, with each pair being formed by one image acquired before the surgery and another after the surgery.

5.1 Experiments with control subjects

In the first set of experiments, we used 20 images from healthy control subjects. For each reference image we created 10 new images by applying random rigid transformations with rotation varying from -20° to 20° around all axes and translations varying from -20 mm to 20 mm along all axes. We then applied the proposed registration method between each of the 20 images and its corresponding 10 created counterparts, performing a total of 200 registration tasks.

The aim of this experiment is to evaluate the registration method in the ideal case, when the moving and fixed images are nearly identical, by assessing its ability to recover the parameters of the applied transformation. Table 5 (first row) shows the mean, standard deviation and maximum values of the rotation and translation errors. It was verified that the registration succeeded on all cases with negligible errors and the method presented a sub-voxel accuracy.

Table 5. Error measurements on the first three sets of experiments (200 image pairs each).

Experiment	Rotation error (Degree)			Translation error (mm)		
	Mean	σ	Max	Mean	σ	Max
Set 1	0.017	0.010	0.083	0.364	0.08	0.720
Set 2	0.018	0.011	0.080	0.356	0.09	0.771
Set 3	0.038	0.040	0.160	0.499	0.09	0.773

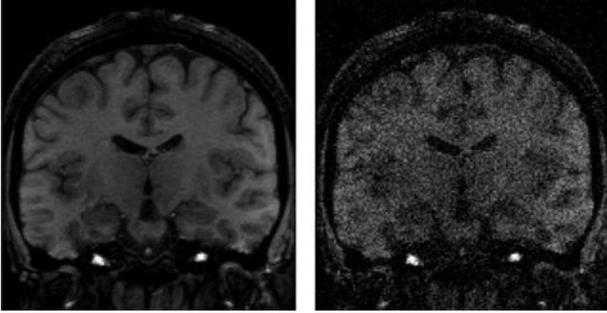


Figure 5. The first image is a control data-set, the second is the same image with noise addition (SNR = 5dB). The degradation of the data is clearly visible.

5.2 Experiments with control subjects and phantom lesions

In the second set of experiments, we explore the performance of the method on images with larger differences simulating the case of surgical tissue removal, which is the focus of our work.

We used the same 20 images from the previous experiment set, but a synthetic artefact mimicking a large surgical cavity was added to the images being registered. Then, we also applied 10 random rigid transformations to each of the 20 images with synthetic lesions and registered them to their respective original control image without lesion, performing a total of 200 registration tasks. The artefacts were added manually with the *Phantom Lesion* filter of the freely available IVS software tool (<http://www.lni.hc.unicamp.br/app/ivs>).

Registration succeeded with negligible errors in all cases as shown on [Table 5](#) (second row).

5.3 Experiments with noisy images

To evaluate the robustness of the method with respect to noise, we used 20 control images and added zero-mean Gaussian noise with SNR=5dB, which represents a high degradation level to the image as seen in [Figure 5](#). The previous experiment (using phantom lesions) was repeated but now with noise addition and, even subject to such a high level of noise, the method performed with no significant accuracy impact, as shown in [Table 5](#) for Set 3.

5.4 Comparison with other methods

Given that registration methods consist of two components, the registration function and the optimiser, the evaluation against other methods in the literature is performed separately for each of these components.

First, we validated our registration function against a widely used and well-known registration function which is

the SSD, computed between voxel intensities from the moving and fixed image, also known as least squares. This method can be extensively used for intra-subject and inter-modality registration because it can be shown that this is the optimum measure when two images only differ by Gaussian noise (Viola 1995; Hill et al. 2001; Hill 2004; Modersitzki 2004; Orchard 2005; Gorbunova 2008). However it is very computationally costly since it uses the whole data from both images to evaluate each candidate match. We repeated the experiment with the images from Set 2 (phantom lesions), keeping the same search algorithm (MSPS) but now using the SSD as registration function to analyse the difference in accuracy in comparison with our registration function and we obtained a mean error of 0.012° (rotation) and 0.11 mm (translation). In comparison with Set 2 on [Table 5](#), although we observe a slightly lower accuracy for our registration function, the difference is so small and the error is still sub-voxel which is acceptable for most applications. Also, in our experiment, the lesioned synthetic images had the same voxel values from the original images (except in the lesion area), which helps the SSD method to perform better. However, in real clinical cases, the differences of intensity will be much stronger, and the accuracy of the SSD method is expected to drop. The same experiment for noisy images (Set 3) revealed a mean error of 0.041° (rotation) and 0.39 mm (translation), which is comparable with our method.

Then, we also evaluate the proposed optimisation algorithm (MSPS) against other optimisation methods. In Section 3 we presented an extensive evaluation of the MSPS algorithm on a testbed of optimisation benchmark problems against other state-of-the-art optimisers, which attested the good performance of the method as a general-purpose black-box optimiser. Now we evaluate the performance of the method specifically for the image registration problem. Again, we repeated the same experiment with Set 2, using the proposed registration function, and we compared the MSPS method with PSO, DE and SA. The results are shown in [Figure 6](#) which reveals that MSPS was able to obtain the best optimisation and also it converged faster than PSO and SA. It also converged faster than DE until 450 iterations.

We have also compared the proposed method with the FMRIB's Linear Image Registration Tool (FLIRT) (<http://fsl.fmrib.ox.ac.uk>), which is part of the widely used open-source software FSL (FMRIB Software Library) from Oxford University. FLIRT is a fully automated, robust and accurate (Jenkinson et al. 2002) tool for brain image affine registration. The tool implements well-known registration cost functions such as mutual information (MI), SSD and correlation ratio (CR). In order to evaluate it, we repeated the experiment with synthetic lesions (Set 2) described in Section 2. We used the recommended FLIRT parameters from Jenkinson et al. (2002) (default for a rigid-body

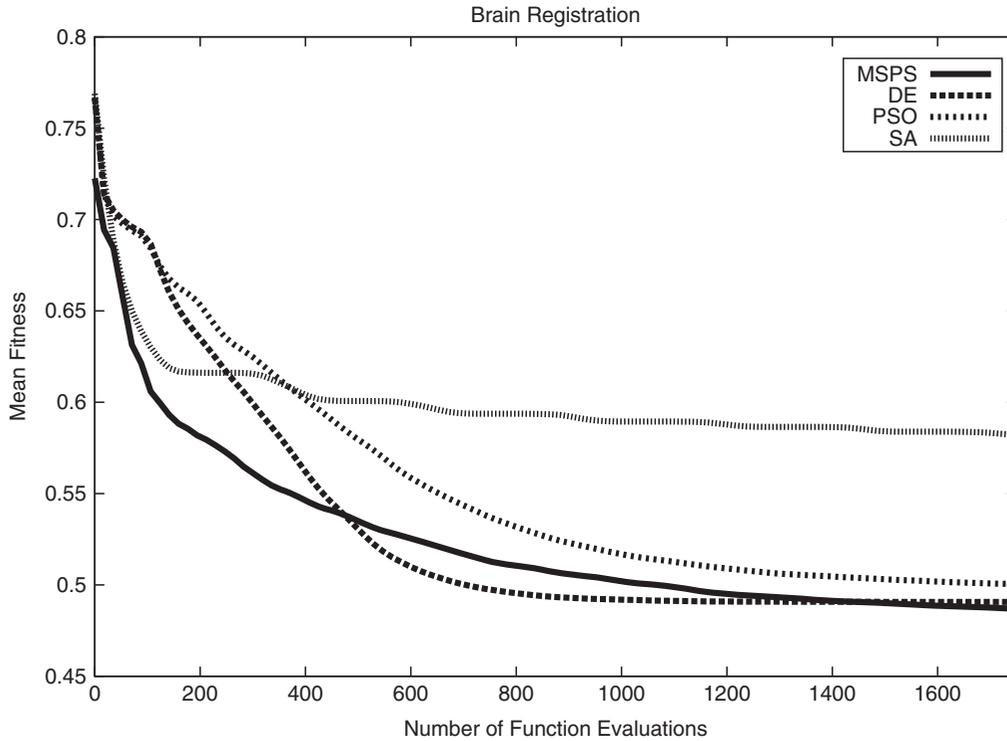


Figure 6. Fitness curve for minimisation of the registration function. The MSPS method presents the best solution and converges faster than PSO and SA.

registration) and evaluated it for three cost functions. The results are shown in Table 6 and reveal that our method presents a very high accuracy, very close to those from FLIRT for all cost functions, which, for practical purposes, can be considered equivalent.

On the other hand, our method is significantly faster than FLIRT. While FLIRT takes on average 20–30 min to complete the task in our computational set-up (described next), our method takes less than 40 s for the same task, a speed-up of 33 times. It is worth mentioning that it is possible to enable a multi-resolution scheme in FLIRT that speeds up the search by beginning from 1/8 scaled-down image, and this scheme drops its computational time to the range of 50–80 s, depending on the cost function. Even without multi-resolution schemes, our method runs faster than FLIRT and we can still speed it up further by using similar multi-resolution schemes.

Table 6. Comparison between the proposed method and FLIRT/FSL for three cost functions: MI, CR and SSD.

Experiment	Rotation error (Degree)			Translation error (mm)		
	Mean	σ	Max	Mean	σ	Max
Proposed method	0.018	0.011	0.080	0.356	0.09	0.771
FLIRT (MI)	0.011	0.007	0.028	0.451	0.23	0.98
FLIRT (CR)	0.010	0.010	0.033	0.449	0.225	0.997
FLIRT (SSD)	0.009	0.007	0.042	0.446	0.226	0.983

The method was implemented in C language and takes on average less than 40 s for a typical MR image ($256 \times 256 \times 160$) running as a single process on a workstation (Intel Xeon E3-1220 3.1 GHz 16GB RAM). Most of this time is spent on the computation of gradient and watershed lines (around 32 s) and the registration itself takes less than 8 s. On the other hand, the SSD-based registration method took on average 15.3 min using the same optimiser (MSPS) and the same number of iterations, which is more than 22 times slower than the proposed method. These results reveal a drastic registration speed-up due to the reduction of key points in the proposed method.

5.5 Experiments with clinical images

To evaluate our method on real clinical images, we used a data-set of 45 pairs (pre- and post-surgery) of MR-T1 images from epilepsy patients, where part of the brain tissue was removed (Figure 8). In this fourth set of experiments, it was not possible to perform a quantitative evaluation, since no ground truths are available. However, a qualitative evaluation was conducted by evaluating the quality of the registration result through visual inspection by an expert using a special visualisation technique that we devised.

A traditional technique to visualise the image registration result is the checkerboard scheme (Ritter

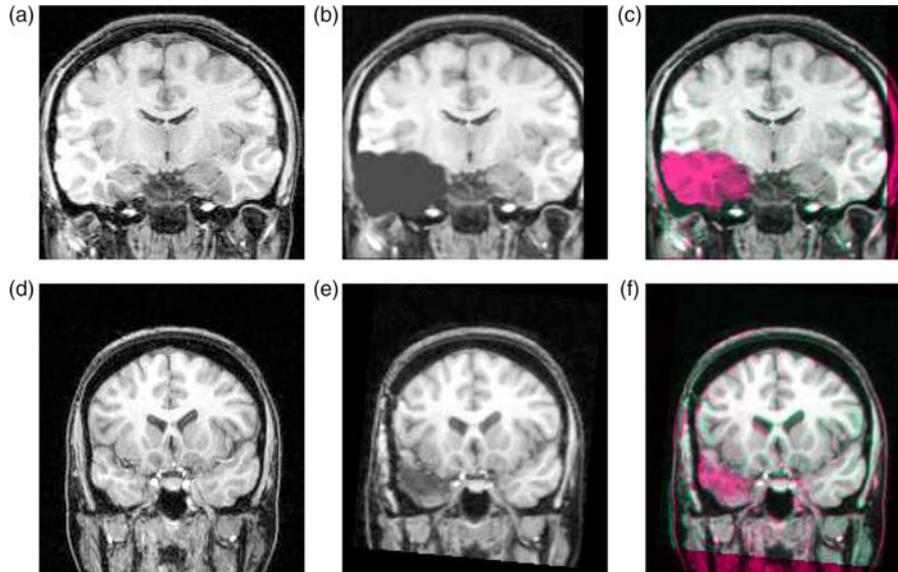


Figure 7. Visualisation of registration using colour coding. The last column shows the combination between the fixed image (first column) and the registered moving image (second column). The first row shows an example of a synthetic lesion case. The second row is an example of clinical image (pre- and post-surgery).

et al. 1999), where the fixed image and the registered image are displayed at the same time in alternated squares like a checkerboard mosaic. However, this scheme does not reveal smaller differences clearly.

We have developed a technique to visualise the result of image registration using colour coding (Ruppert et al. 2010). In this scheme, both images are normalised and then combined by assigning the red channel to one image and the green channel to the other. The blue channel is filled with the average of both images. This way, the regions with accurate registration appear in unsaturated grey, while regions of intensity mismatches appear in red- or green-hued voxels. When the registration is accurate, few saturated voxels are seen outside the surgery area. Naturally, clinical aspects have to be taken into account, such as the swelling of regions around the removed tissue. Figure 7 illustrates this technique on a synthetic lesion and on a clinical image.

By using this method, the visual inspection of all 45 cases by a neurosurgeon expert showed that the registration method performed successfully on all cases, revealing just the operated region in red- or green-hued voxels, while the whole image is seen as unsaturated grey voxels (Figure 8).

5.6 Computational time

The method was implemented in C language and takes on average less than 40 s for a typical MR image ($256 \times 256 \times 160$) running as a single process on a workstation (Intel Xeon E3-1220 3.1GHz 6GB RAM). Most of this time is spent on the computation of gradient

and watershed lines (around 32 s) and the registration itself takes less than 8 s. On the other hand, the SSD-based registration method took on average 15.3 min using the same optimiser (MSPS) and the same number of iterations, which is more than 22 times slower than the proposed method. These results reveal a drastic registration speed-up due to the reduction of key points in the proposed method.

6. Conclusion and future work

We presented a method for fast and automatic 3D medical image registration based on the watershed transform from the greyscale marker and the MSPS algorithm. The method was evaluated for rigid registration of 3D MR-T1 images, given our major interest in the analysis of pre- and post-surgery images from epilepsy patients. The watershed transform from the greyscale marker was able to reduce the subset of key points for registration and yet maintain high accuracy. The experiments involved 645 registration tasks with images of control subjects and patients, before and after surgery, and images with synthetic lesions and noise. The results showed that the method is much faster with similar accuracy than other methods. The considerable speed-up (22× to 33× faster) is especially important in emergency cases, when the treatment performed a few minutes earlier represents a significant clinical outcome difference. Also, the growing use of intra-operative MRI requires faster registration algorithms to reduce surgery time, risks and costs. Also, the colour-coding visualisation scheme was shown to be a valuable aid for qualitative evaluation of the registration results.

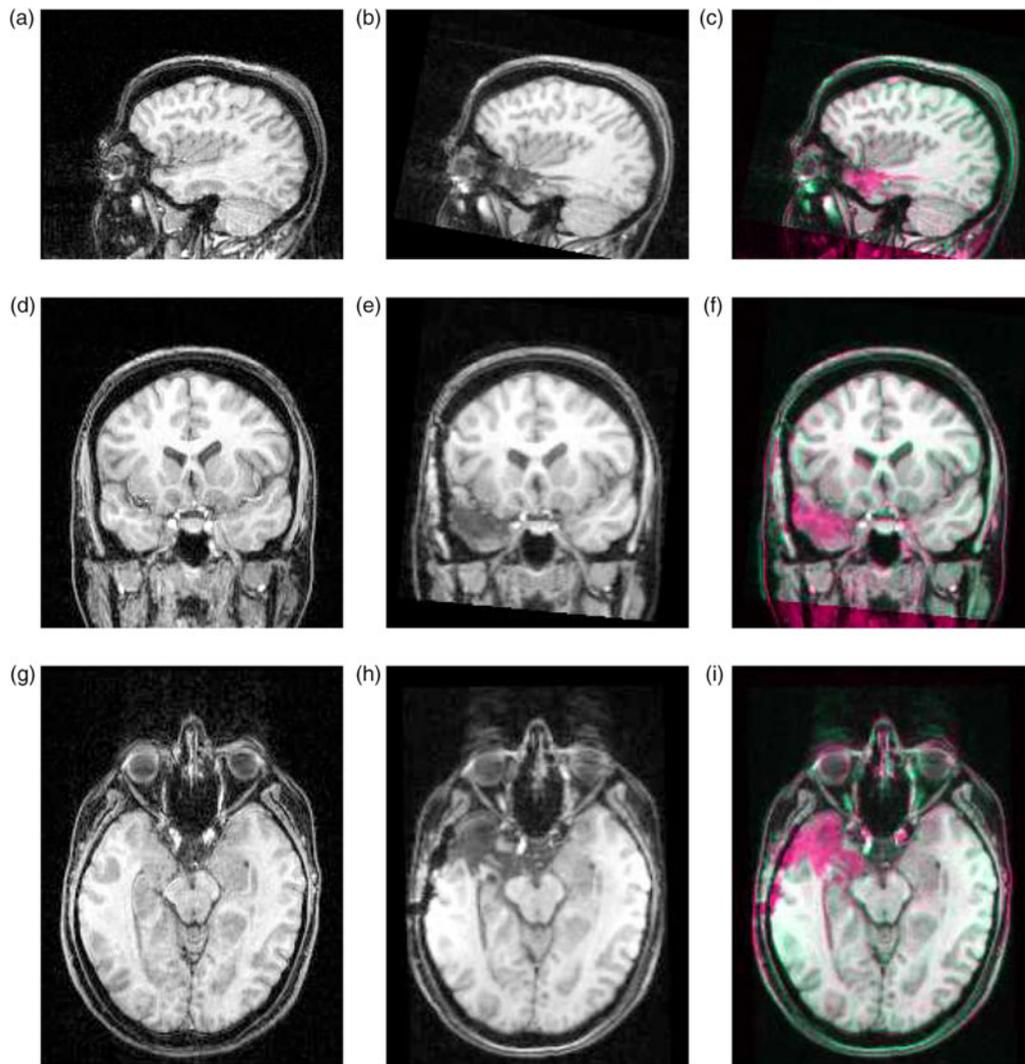


Figure 8. Slices of a clinical image (pre- and post-surgery). Part of the brain removed by surgery can be seen in red colour.

A key contribution is the new version of the MSPS, which was extensively evaluated in comparison with other four widely used optimisers. The major contestants were PSO and DE, the methods whose good performance is of common knowledge in the literature. We carried out a fair evaluation using meta-optimisation on a difficult testbed of 12 benchmark problems. We obtained promising results in these benchmark problems, with MSPS outperforming the other methods in most cases. Especially, MSPS performed best in all of the considered non-convex problems. This allows us to conclude that several other problems can also be well addressed with this technique, opening avenues to explore MSPS in other applications.

Future work includes comparison with other feature-based methods, extensions of the method to other imaging modalities and to global deformable registration with new image features, transformation models and search parameters. We also plan to apply the MSPS method to

other real-world problems, mainly in the fields of medical imaging and computer vision.

Acknowledgement

We are also grateful to the Neuroimaging Laboratory of the University of Campinas for providing the MR images used in this work.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The authors thank FAPESP [grant number 2007/52015-0], [grant number 2010/00994-8], [grant number 2010/05647-4] and [grant number 2013/11359-0]; CNPq [grant number 302970/2014-2],

[grant number 477662/2013-7] and [grant number 304352/2012-8], [grant number 479070/2013-0], [grant number 304352/2012-8], and [grant number 477662/2013-7]; CAPES DeepEyes project and Microsoft for the financial support.

References

- Allaire G. 2007. Numerical analysis and optimization. Oxford: Oxford University Press. Numerical Mathematics and Scientific Computation.
- Audette M, Ferrie F, Peters T. 2000. An algorithmic overview of surface registration techniques for medical imaging. *Med Image Anal.* 4(3):201–217. doi:10.1016/S1361-8415(00)00014-1.
- Auger A, Hansen N, Zerpa JMP, Ros R, Schoenauer M. 2009. Experimental comparisons of derivative free optimization algorithms. In: Vahrenhold J, editor. 8th international symposium on experimental algorithms. LNCS. Vol. 5526 Berlin: Springer; p. 3–15.
- Bergo FPG, Falcão AX, Miranda PAV, Rocha LM. 2007. Automatic image segmentation by tree pruning. *J Math Imaging Vision.* 29(2–3):141–162. doi:10.1007/s10851-007-0035-4.
- Birattari M, Dorigo M. 2007. How to assess and report the performance of a stochastic algorithm on a benchmark problem: mean or best result on a number of runs? *Optim Lett.* 1(3):309–311. doi:10.1007/s11590-006-0011-8.
- Conn AR, Scheinberg K, Vicente LN. 2009. Introduction to derivative-free optimization. Philadelphia, PA: MPS/SIAM.
- Crum WR, Hartkens T, Hill DLG. 2004. Non-rigid image registration: theory and practice. *Br J Radiol.* 77(Suppl_2): S140–S153. doi:10.1259/bjrr/25329214.
- da Silva A, dos Santos J, Falcão A, Torres R, Magalhães L. 2012. Incorporating multiple distance spaces in optimum-path forest classification to improve feedback-based learning. *Comput Vision Image Understanding.* 116(4):510–523. doi:10.1016/j.cviu.2011.12.001.
- De Miranda PA, Falcão AX, Udupa JK. 2010. Synergistic arc-weight estimation for interactive image segmentation using graphs. *Comput Vision Image Understanding.* 114(1):85–99. doi:10.1016/j.cviu.2009.08.001.
- Falcão AX, Stolfi J, Lotufo RA. 2004. The image foresting transform: theory, algorithms and applications. *IEEE Trans Pattern Anal Mach Intell.* 26(1):19–29.
- Feldmar J, Malandain G, Declerck J, Ayache N. 1996. Extension of the ICP algorithm to non-rigid intensity-based registration of 3D volumes. In: *Math Methods Biomed Image Anal.* Los Alamitos, CA: IEEE Computer Society Press; p. 84–93.
- Floudas CA, Gounaris CE. 2009. A review of recent advances in global optimization. *J Global Optim.* 45(1):3–38. doi:10.1007/s10898-008-9332-8.
- Gorbunova V, et al. 2008. Weight preserving image registration for monitoring disease progression in lung ct. In: *Proceedings of Miccai 2008.* Berlin: Springer; p. 863–870.
- Goshtasby AA. 2012. Image registration: principles, tools and methods. London: Springer.
- Gray P, Hart W, Painton L, Phillips C, Trahan M, Wagner J. 1997. A survey of global optimization methods. Sandia National Laboratories. Report no. Consulted April 2012. <http://www.cs.sandia.gov/opt/survey/main.html>
- Hajnal J, Hill DLG, Hawkes DJ. 2001. Medical image registration. London: CRC Press.
- Hallpike L, Hawkes DJ. 2002. Medical image registration: an overview. *Imaging.* 14(6):455–463. doi:10.1259/img.14.6.140455.
- Han J, Qiao M, Hornegger J, Kuwert T, Bautz W, Romer W. 2006. Automatic sub-volume registration by probabilistic random search. *Med Imaging – Image Process – Proc SPIE.* 6144:799–807.
- Hemker T. 2008. Derivative free surrogate optimization for mixed-integer nonlinear black box problems in engineering [dissertation]. Technische Universität Darmstadt.
- Hill DJH. 2004. Across-modality registration using intensity-based cost functions. In: *Handbook of medical imaging.* San Diego, CA: Academic Press; p. 537–553.
- Hill DLG, Batchelor PG, Holden M, Hawkes DJ. 2001. Medical image registration. *Phys Med Biol.* 46(3):R1–R45. doi:10.1088/0031-9155/46/3/201.
- Huang C, Jiang C, Sung W. 2006. Medical image registration and fusion with 3D CT and MR data of head. *Computer-Based Med Syst.* 401:404.
- Huang D, Allen TT, Notz WI, Zeng N. 2006. Global optimization of stochastic black-box systems via sequential kriging meta-models. *J Global Optim.* 34(3):441–466. doi:10.1007/s10898-005-2454-3.
- Jenkinson M, Bannister P, Brady JM, Smith SM. 2002. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *NeuroImage.* 17(2):825–841. doi:10.1006/nimg.2002.1132.
- Kennedy J, Eberhart R. 2002. Particle swarm optimization. In: *Neural Networks, 1995. Proceedings, IEEE International Conference on August, Perth, WA.* Vol. 4. Piscataway, NJ: IEEE; p. 1942–1948.
- Kirkpatrick S, Gelatt CD, Vecchi MP. 1983. Optimization by simulated annealing. *Science.* 220(4598):671–680.
- Kostelec PJ, Periaswamy S. 2003. Image registration for MRI. *Modern Signal Process.* 46:161–184.
- Lee J. 2007. A novel three-phase trajectory informed search methodology for global optimization. *J Global Optim.* 38(1):61–77. doi:10.1007/s10898-006-9083-3.
- Lotufo R, Falcão A, Zampiroli F. 2002. IFT-Watershed from gray-scale marker. In: *Proc. of xv brazilian symp. on computer graphics and image processing.* Piscataway, NJ: IEEE; p. 146–152.
- Lu Z, Feng Q, Shi P, Chen W. 2007. A fast 3-D medical image registration algorithm based on equivalent meridian plane. *Proc IEEE Int Conf Image Process.* 5:357–360.
- Maes F, Collignon A, Vandermeulen D, Marchal G, Suetens P. 1997. Multimodality image registration by maximization of mutual information. *IEEE Trans Med Imaging.* 16(2):187–198. doi:10.1109/42.563664.
- Maintz JBA, Viergever MA. 1998. A survey of medical image registration. *Med Image Anal.* 2(1):1–36. doi:10.1016/S1361-8415(01)80026-8.
- Miranda PA, Falcao AX, Spina TV. 2012. Riverbed: a novel user-steered image segmentation method based on optimum boundary tracking. *Image Process IEEE Trans.* 21(6):3042–3052. doi:10.1109/TIP.2012.2188034.
- Modersitzki J. 2004. Numerical methods for image registration. Oxford: Oxford University Press.
- Orchard J. 2005. Efficient global weighted least-squares translation registration in the frequency domain. In: *Image Analysis and Recognition – Lecture Notes in Computer Science 2005.* ICIAR 2005, Toronto, Canada. Vol. 3656. p. 116–124.
- Pedersen MEH. 2008. SwarmOps: Black-Box Optimization in ANSI C. Available from: <http://www.hvass-labs.org/projects/swarmops/c/>
- Pedersen MEH. 2010. Tuning & simplifying heuristical optimization [dissertation]. University of Southampton.

- Pietrzyk U, Herholz K, Fink G, Jacobs A, Mielke R, Slansky I, Wurker M, Heiss WD. 1994. An interactive technique for three-dimensional image registration: validation for PET, SPECT, MRI and CT brain studies. *J Nucl Med.* 35(12):2011–2018.
- Rardin RL, Uzsoy R. 2001. Experimental evaluation of heuristic optimization algorithms: a tutorial. *J Heuristics.* 7(3):261–304. Available from: <http://www.springerlink.com/index/T63RR6351V832Q17.pdf>
- Rashedi E, Nezamabadi-pour H, Saryazdi S. 2009. GSA: a gravitational search algorithm. *Inf Sci.* 179(13):2232–2248. doi:10.1016/j.ins.2009.03.004.
- Ritter N, Owens R, Cooper J, Eikelboom R, van Saarloos PP. 1999. Registration of stereo and temporal images of the retina. *IEEE Trans Med Imaging.* 18(5):404–418. doi:10.1109/42.774168.
- Rueckert D, Aljabar P. 2010. Nonrigid registration of medical images: theory, methods, and applications [applications corner. *Signal Process Magazine, IEEE.* 27(4):113–119. doi:10.1109/MSP.2010.936850.
- Rueckert D, Schnabel JA. 2011. Medical image registration. In: *Biomedical image processing Chapter 5.* Berlin: Springer; p. 131–154.
- Ruppert GCS, Favretto F, Falcão AX, Yasuda CL, Bergo FPG. 2010. Fast and accurate image registration using the multiscale parametric space and grayscale watershed. In: *Proc. of the international conference on systems, signals, and image processing (IWSSIP).* Niteroi, RJ: EdUFF; p. 457–460.
- Shams R, Sadeghi P, Kennedy RA, Hartley RI. 2010. A survey of medical image registration on multicore and the gpu. *IEEE Signal Process Mag.* 27(2):50–60. doi:10.1109/MSP.2009.935387.
- Smit SK, Eiben AE. 2009. Comparing parameter tuning methods for evolutionary algorithms. In: *Proc. of 11th conference on congress on evolutionary computation.* Piscataway, NJ: IEEE Press; CEC'09 p. 399–406.
- Sotiras A, Davatzikos C, Paragios N. 2013. Deformable medical image registration: a survey. *IEEE Trans Med Imaging.* 32(7):1153–1190. doi:10.1109/TMI.2013.2265603.
- Spina TV, Montoya-Zegarra JA, Falcao AX, Miranda P. 2009. Fast interactive segmentation of natural images using the image foresting transform. In: *Digital Signal Processing, 2009 16th International Conference on.* Piscataway, NJ: IEEE; p. 1–8.
- Storn R, Price K. 1997. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *J Global Optim.* 11(4):341–359. doi:10.1023/A:1008202821328.
- Sun W, Dong Y. 2011. Study of multiscale global optimization based on parameter space partition. *J Global Optim.* 49(1):149–172. doi:10.1007/s10898-010-9540-x.
- Vandermeulen D, Vandermeulen F, Suetens P. 1999. Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information. *Med Image Anal.* 3(4):373–386. doi:10.1016/S1361-8415(99)80030-9.
- Viola P, Wells, III, WM. 1995. Alignment by maximization of mutual information. *Int J Comput Vision.* 24(2):137–154. doi:10.1023/A:1007958904918.
- Viola PA. 1995. Alignment by maximization of mutual information [dissertation]. Ph.D. thesis, Massachusetts Institute of Technology.
- Wen HH, Lin WC, Chen CT. 1996. Knowledge-based medical image registration. *Eng Med Biol Soc.* 3(1200):1201.
- Wen P. 2008. Medical image registration based-on point, contour and curves. In: *Proceedings of the international conference on biomedical engineering and informatics, BMEI 2008.* Piscataway, NJ: IEEE; p. 132–136.
- Wolpert DH, Macready WG. 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation.* 1(1):67–82. doi:10.1109/4235.585893.
- Xin yao X, Yong liu Y, Guangming lin G. 1999. Evolutionary programming made faster. *Evolutionary Computation, IEEE Transactions on.* 3(2):82–102. doi:10.1109/4235.771163.
- Yasuda CL, Valise CC, Saúde AV, Costa ALF, Pereira FR, Morita M, Betting LE, Tedeschi H, Oliveira E, Castellano G, Cendes F. 2008. Recovery of white matter atrophy (WMA) after successful surgery in mesial TLE. In: *Proc. 60th annual meeting of the american academy of neurology.* Chicago, IL; p. A4.
- Zhang Y, Liang X, Ma J, Jing Y, Gonzales MJ, Villongco C, Krishnamurthy A, Frank LR, Nigam V, Stark P, et al. 2012. An atlas-based geometry pipeline for cardiac hermite model construction and diffusion tensor reorientation. *Med Image Anal.* 16(6):1130–1141. doi:10.1016/j.media.2012.06.005.
- Zitová B, Flusser J. 2003. Image registration methods: a survey. *Image Vision Comput.* 21(11):977–1000. doi:10.1016/S0262-8856(03)00137-9.
- Zou X, Zhao X, Feng Y. 2007. An efficient medical image registration algorithm based on gradient descent. *Complex Med Eng.* 1:636–639.