Proc. of SPIE Vol. 3169, Wavelet Applications in Signal and Image Processing V, ed. A Aldroubi, A F Laine, M A Unser (Oct 1997) Copyright SPIE

Spline Pyramids for Inter-Modal Image Registration Using Mutual Information

Philippe Thévenaz and Michael Unser

BEIP/National Center for Research Resources
National Institutes of Health, Bethesda MD 20892-5766, USA

ABSTRACT

We propose a new optimizer for multiresolution image registration. It is adapted to a criterion known as mutual information and is well suited to inter-modality. Our iteration strategy is inspired by the Marquardt-Levenberg algorithm, even though the underlying problem is not least-squares. We develop a framework based on a continuous polynomial spline representation of images. Together with the use of Parzen histogram estimates, it allows for closed-form expressions of the gradient and Hessian of the criterion. Tremendous simplifications result from the choice of Parzen windows satisfying the partition of unity, also based on B-splines. We use this framework to compute an image pyramid and to set our optimizer in a multiresolution context. We perform several experiments and show that it is particularly well adapted to a coarse-to-fine optimization strategy. We compare our approach to the popular Powell algorithm and conclude that our proposed optimizer is faster, at no cost in robustness or precision.

Keywords: Multiresolution, B-Spline, Parzen window, Marquardt-Levenberg

1. INTRODUCTION

Image registration addresses the following problem: given two images, find a geometric transformation that maps the first image into the second one. This problem is often encountered in biomedical applications where it can take two basic forms: in one case, the two images to be aligned exhibit only a small structural discrepancy, as arises when their sole difference is the condition of the subject (*e.g.*, resting versus performing a task). In this case, we speak of intra-modal registration. Alternatively, in the case of inter-modal registration, the subject is imaged in essentially two different ways (*e.g.*, image one might be the result of measuring his local glucose uptake in the brain, and image two might be the result of measuring a proton density). The task of registering the two images becomes more complicated because there might be no direct relation between the intensities of the two images.

Several ways have been devised to solve this problem.[1,2] In the past, many authors have relied on extracting geometric features that are observable in both modalities, and have attempted to register the images based on these features. Very often, the selected features were a particular set of points, curves or surfaces. Recently, a new type of solution has emerged,[3,4] based on the use of the mutual information between the two images. The basic idea is to apply a criterion that measures their *statistical dependence*; the value of this criterion is then optimized in the sense that one seeks the geometric transformation that maximizes the dependence between the two images. The advantage of the new approach over the older ones is that now every pixel contributes to the criterion, which makes it more robust. In addition, it is also more general, because there is no need for the *a priori* knowledge needed previously in order to perform the segmentation into feature/non-feature map.

In this paper, we develop a new algorithm for the optimization of the above criterion. It is designed to perform best in a multiresolution context, and takes advantage of a coherent framework for the representation of continuous images. This allows explicit expressions for the gradient of the criterion. Our optimizer is inspired by the Marquardt-Levenberg strategy for general non-linear least-squares problems. We note, however, that our method is specific because our criterion does not involve least-squares. Experiments show that our new algorithm improves the registration speed and accuracy.

This paper is organized as follows: in Section 2., we describe the mutual information criterion and we design a way to achieve its computation. In Section 3., we introduce a continuous model for the images and we discuss the benefits of a multiresolution approach. In Section 4., we describe a new algorithm that optimizes the mutual information between the two images to register. In Section 5., we present some experiments and compare several approaches.

2. MUTUAL INFORMATION

The mutual information between two images can be regarded a statistical tool to measure the degree to which one image can be predicted from the other. In this paper, we specifically deal with the Kullback-Leibler measure that was recently proposed as a registration criterion by several authors.[4,5]

2.1. Definitions

Let $f_T(\mathbf{x})$ be a test image we want to align to a reference image $f_R(\mathbf{x})$. These images are defined on a continuous domain $\mathbf{x} \in V$ that may have any number of dimensions (*e.g.*, surface, volume). The coordinates $\mathbf{x}_i$ are samples of V. Let $\mathbf{g}(\mathbf{x}; \mu_1, \mu_2, \ldots)$ be some geometric transformation with associated parameters $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots)$. The problem of image registration is to find the set of parameters $(\hat{\mu}_1, \hat{\mu}_2, \ldots)$ that brings the transformed test image $f_T(\mathbf{g}(\mathbf{x}))$ into best correspondence with the reference image.

Let L_T and L_R be discrete sets of intensities associated to the test and the reference image, respectively. Let $w(\xi)$ be a separable Parzen window satisfying $w(\xi) \geq 0$, $\forall \xi$. We define the joint discrete Parzen probability as

$$p(\iota, \kappa; \boldsymbol{\mu}) = \alpha(\boldsymbol{\mu}) \sum_{\mathbf{x}_i \in V} w(\iota - f_T(\mathbf{g}(\mathbf{x}_i; \boldsymbol{\mu}))) \, w(\kappa - f_R(\mathbf{x}_i)), \tag{1}$$

where α is a normalization factor that ensures $\sum p(\iota, \kappa) = 1$, and where $\iota \in L_T$ and $\kappa \in L_R$. The marginal discrete probabilities are given by

$$p_T(\iota; \boldsymbol{\mu}) = \sum_{\kappa \in L_R} p(\iota, \kappa; \boldsymbol{\mu}), \tag{2}$$

$$p_R(\kappa; \boldsymbol{\mu}) = \sum_{\iota \in L_T} p(\iota, \kappa; \boldsymbol{\mu}). \tag{3}$$

The negative of the mutual information S between the transformed test image and the reference image is

$$S(\boldsymbol{\mu}) = - \sum_{\iota \in L_T} \sum_{\kappa \in L_R} p(\iota, \kappa; \boldsymbol{\mu}) \log_2 \frac{p(\iota, \kappa; \boldsymbol{\mu})}{p_T(\iota; \boldsymbol{\mu}) \, p_R(\kappa; \boldsymbol{\mu})}. \tag{4}$$

It might happen that some of the terms $p(\iota, \kappa)$ take a zero value. For example, when the two images come from the same modality, the best registration that one may achieve, based on the above criterion, is one in which the only non-zero terms are those on the diagonal of the matrix p. However, vanishing entries in the joint discrete Parzen probability are to be ignored when computing the mutual information criterion because of the well known relation $\lim_{x \to 0^+} x \log x = 0$.

2.2. Parzen window

An unfortunate consequence of computing p_R as a marginal probability is that it makes it depend explicitly on the transformation parameters $(\mu_1, \mu_2, \ldots)$. Although the reference image doesn't change with a variation in these parameters, p_R is sensitive to them because of the coupling introduced through the separable Parzen window w. One way to avoid this effect is to introduce the partition of unity constraint

$$\sum_{\xi \in L} w(x + \xi) = 1, \ \forall x. \tag{5}$$

When this constraint is satisfied for any fixed value x, it is easy to show that the marginal probability p_R becomes independent of the transformation parameters $(\mu_1, \mu_2, \ldots)$

$$p_R(\kappa) = \alpha \sum_{\mathbf{x} \in V} w(\kappa - f_R(\mathbf{x})), \ \forall (\mu_1, \mu_2, \ldots). \tag{6}$$

Another advantage is that the normalization factor α now takes a constant value. A very simple Parzen window that satisfies the partition of unity constraint is a centered unit square pulse. In this particular case, for any x, and for L a set of integers, exactly one term contributes to the sum in (5). Meanwhile, the joint probability $p(\iota, \kappa)$ defined by (1) behaves like a traditional frequency histogram.

2.3. B-splines

B-spline functions have many interesting properties.[6,7] Of particular relevance for this paper is the fact that they satisfy the constraint for the partition of unity (5). In addition, they have the advantage of being smooth functions with explicit derivatives. The B-spline $\beta^{(n)}$ is a piece-wise polynomial of integer degree $n \geq 0$ that can be recursively defined by the following convolution:

$$\beta^{(n)}(x) = (\beta^{(n-1)} * \beta^{(0)})(x) = \int_{-\infty}^{\infty} \beta^{(n-1)}(x)\, \beta^{(0)}(x-t)\, \mathrm{d}t,\ n > 0, \tag{7}$$

where $\beta^{(0)}$ is a unit square pulse

$$\beta^{(0)}(x) = \begin{cases} 1 & x \in [-1/2, 1/2) \\ 0 & x \notin [-1/2, 1/2) \end{cases}. \tag{8}$$

Not only will these B-splines be used as Parzen window, but also they will provide the basis functions for representing continuous images given by a set of samples.

3. MULTIRESOLUTION

Multiresolution is an optimization strategy well suited to problems for which the task is to improve on a given current solution (or initial condition). The problem is first solved on a coarse scale, with few data, where it quickly yields an approximate solution. The scale is then refined, more data are taken into account, and the solution from the coarse scale is propagated to the finer scale. This process is iterated until the finest scale is reached.

3.1. Image model

Let us assume that an image $f(\mathbf{x})$ is known through a set of samples $f_i = f(\mathbf{x}_i)$ that are regularly spaced on a Cartesian grid. To be useful, an image model must satisfy several constraints. First, it must allow one to interpolate an image, which provides the link between, on one hand, the samples f_i and their location $\mathbf{x}_i$, and on the other hand, the continuous function $f(\mathbf{x})$. This property is typically needed when performing the geometric transformation $f \to f(\mathbf{g}(\mathbf{x}_i))$. Second, given some continuous function $y(\mathbf{x})$, there must exist a procedure to recover a set of samples y_i at locations $\mathbf{x}_i$ such that the model based on this set would reconstruct a close approximation to $y(\mathbf{x})$. A typical application of this requirement arises when one computes a resolution pyramid, for in this case the procedure can be sketched by $(f_i, \mathbf{x}_i) \to f(\mathbf{x}) \to f(2\mathbf{x}) = y(\mathbf{x}) \to (y_i, \mathbf{x}_i)$.

We base our image model on the B-spline functions of degree n introduced at Section 2.3. More explicitly, we have

$$f(\mathbf{x}) = \sum_i c(\mathbf{x}_i)\, \beta^{(n)}(\mathbf{x} - \mathbf{x}_i), \tag{9}$$

where $\beta(\mathbf{x})$ is a separable convolution kernel given by the product $\beta^{(n)}(x_1) \cdot \beta^{(n)}(x_2) \cdot \cdots$, and where the expansion B-spline coefficients c are computed from the sample values f_i by recursive digital filtering.[7] This model is continuous, differentiable a.e. (almost everywhere) for $n \geq 0$, and differentiable for $n > 1$. It serves three purposes. First, its rescaled versions yield the image pyramid that we use for our multiresolution approach.[8] Second, it allows us to re-sample the transformed image $f(\mathbf{g}(\mathbf{x}_i))$. Finally, it is used in computing the image gradient needed during optimization.

3.2. Model degree

The model degree determines the quality of the approach. The lowest possible degree $n = 0$ is called nearest-neighbor. Used to compute the resolution pyramid, it results in severe aliasing. Used to compute $f(\mathbf{g}(\mathbf{x}))$, it results in blocking artifacts. Used to compute S, it results in a discontinuous criterion, which is hard to optimize. Also, the optimum is generally not uniquely defined. The next degree $n = 1$ corresponds to linear interpolation. It results in less aliasing, and oversmoothing substitutes for blocking. Meanwhile, the criterion is better-behaved. In these two cases, the computation of the B-spline coefficients c is trivial. For higher degrees easy computation is no longer the case, but aliasing is reduced substantially. Blocking and smoothing are gradually replaced by ringing. At the extreme, when $n \to \infty$, aliasing disappears altogether but ringing is strongly present (sinc, or Shannon interpolation[9]). A good compromise between all these issues is to select a cubic B-spline $\beta^{(3)}$ as model kernel.

There are three major reasons why the choice of a high-quality model is essential to the proper behavior of a multiresolution registration method. First, consider performing optimization at a coarse level of the pyramid. The steps made by the optimizer at this level correspond to big strides at the finest level. It follows that precision is of utmost importance at this coarse level, and subpixel interpolation must be faithful. This calls for a degree n that is higher than what is traditionally selected. Second, consider having found the optimal parameter $\hat{\boldsymbol{\mu}}$ at some level l. The optimal parameter at the next finer level $l+1$ is not identical because data are more detailed, and the added details call for some corrective action. It is however desired that the corrections be as small as possible, which is achieved by minimizing the amount of detail distinguishing level l from level $l+1$. Thus, it is best to limit the aliasing inherent in the size reduction operation, which again calls for a high model degree n. We shall see in Section 4. that our optimizer requires a differentiable kernel. We prefer to avoid having to sample a derivative where it is discontinuous, which could sometimes arise with linear interpolation. This is one more reason to select a high model degree n.

3.3. Robustness

Another well-known benefit of using a resolution pyramid is that it usually strengthens the robustness of the optimization algorithm. The mechanism is as follows: when data are coarse, the loss of image detail often results in a reduction of the details in the criterion itself. Thus, local optima in which optimization algorithms might be trapped tend to disappear.

3.4. Speed

In addition to optimizing at the coarse levels, the multiresolution strategy does not preclude optimization at the finest one. For this strategy to be efficient in terms of computation time, it is required that, at the finest level, the number of iterations necessary to reach some registration precision be less than the number needed to solve the same problem without a multiresolution strategy. From this consideration, it follows that it is important to select an optimizer that benefits strongly from good starting conditions. As examples of bad candidates, one can think of many direction-set methods (*e.g.*, conjugate-gradient with or without explicit derivatives), where the optimizer often needs to explore several directions in the $(\mu_1, \mu_2, \ldots)$ space sequentially, before even starting to really optimize. With such algorithms, especially when the conditions are nearly optimal, many criterion evaluations are wasted simply to assess that the conditions are, well, nearly optimal. We prefer to use optimizers that have a global 'understanding' of their immediate surroundings, since it is likely that the optimum for which we search will be close. It is even better if the convergence were to be superlinear, a regime in which the optimizer converges quadratically (or better) when the optimum is close enough. The goal of the next Section is to present such an optimizer.

4. OPTIMIZATION

We present now the main contribution of this paper. It is an optimization algorithm based on the same strategy as that of the Marquardt-Levenberg optimizer.[10] An important difference is that our specific registration problem is non least-squares. Our optimizer is iterative; it proceeds by trying potentially better solutions around a given initial condition. Apart from the propagation of the final solution from a coarser level to the next finer level, where it will be used as initial condition, the existence of an underlying image pyramid is ignored while optimizing within any given level. Hence, we present this algorithm out of the multiresolution context.

4.1. Criterion model

As a first step, let us express the mutual information (4) by a Taylor expansion

$$S(\boldsymbol{\mu}) = S(\boldsymbol{\nu}) + \sum_i \frac{\partial S(\boldsymbol{\nu})}{\partial \mu_i} (\mu_i - \nu_i) + \frac{1}{2} \sum_{i,j} \frac{\partial^2 S(\boldsymbol{\nu})}{\partial \mu_i \, \partial \mu_j} (\mu_i - \nu_i)(\mu_j - \nu_j) + \cdots . \tag{10}$$

We then simplify (10) by ignoring all terms above second-order. If both $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are not too far from the optimum, this quadratic model is known to be quite appropriate.

4.2. Gradient

Let us define the gradient $\boldsymbol{\Delta} S$ as

$$\boldsymbol{\Delta} S = \left[\frac{\partial S}{\partial \mu_1}, \frac{\partial S}{\partial \mu_2}, \ldots\right]^{\mathsf{T}}. \tag{11}$$

Based on (4) and (1), and selecting a B-spline of degree m as a Parzen window satisfying the partition of unity condition, we can determine a component of the gradient as

$$\frac{\partial S}{\partial \mu} = -\sum_{\iota \in L_T} \sum_{\kappa \in L_R} \frac{\partial p(\iota, \kappa)}{\partial \mu} \log_2 \frac{p(\iota, \kappa)}{p_T(\iota)}, \tag{12}$$

where the gradient of the joint probability distribution is expanded as

$$\frac{\partial p(\iota, \kappa)}{\partial \mu} = \frac{1}{(\#V)} \sum_{\mathbf{x} \in V} \beta^{(m)}(\kappa - f_R(\mathbf{x})) \left.\frac{\partial \beta^{(m)}(\xi)}{\partial \xi}\right|_{\xi = \iota - f_T(\mathbf{g}(\mathbf{x};\boldsymbol{\mu}))} \left(\left.\frac{-\mathrm{d}f_T(\mathbf{t})}{\mathrm{d}\mathbf{t}}\right|_{\mathbf{t}=\mathbf{g}(\mathbf{x};\boldsymbol{\mu})}\right)^{\mathsf{T}} \frac{\partial \mathbf{g}(\mathbf{x};\boldsymbol{\mu})}{\partial \mu}, \tag{13}$$

where it is possible to introduce the explicit expression for the derivative of a B-spline derived from (7)

$$\frac{\partial \beta^{(m)}(\xi)}{\partial \xi} = \beta^{(m-1)}(\xi + 1/2) - \beta^{(m-1)}(\xi - 1/2), \tag{14}$$

and where the spatial gradient of an image $\mathrm{d}f(\mathbf{t})/\mathrm{d}\mathbf{t}$ is given through the B-spline model of degree n

$$\frac{\mathrm{d}f_T(\mathbf{t})}{\mathrm{d}\mathbf{t}} = \sum_i c(\mathbf{x}_i) \left.\frac{\mathrm{d}\beta^{(n)}(\mathbf{u})}{\mathrm{d}\mathbf{u}}\right|_{\mathbf{u}=\mathbf{t}-\mathbf{x}_i} = \sum_i c(\mathbf{x}_i) \begin{bmatrix} \left.\frac{\partial \beta^{(n)}(u)}{\partial u}\right|_{u=(\mathbf{t}-\mathbf{x}_i)_1} \beta^{(n)}((\mathbf{t}-\mathbf{x}_i)_2) \cdots \\ \beta^{(n)}((\mathbf{t}-\mathbf{x}_i)_1) \left.\frac{\partial \beta^{(n)}(u)}{\partial u}\right|_{u=(\mathbf{t}-\mathbf{x}_i)_2} \cdots \\ \vdots \end{bmatrix}. \tag{15}$$

The last unexplained term in (13) is $\partial \mathbf{g}(\mathbf{x};\boldsymbol{\mu})/\partial \mu$, which describes the variation in position due to a variation in parameter. This term depends on geometry alone. Finally, the gradient of the marginal joint density can be expressed by

$$\frac{\partial p_T(\iota)}{\partial \mu} = \sum_{\kappa \in L_R} \frac{\partial p(\iota, \kappa)}{\partial \mu} = \frac{1}{(\#V)} \sum_{\mathbf{x} \in V} \left.\frac{\partial \beta^{(m)}(\xi)}{\partial \xi}\right|_{\xi = \iota - f_T(\mathbf{g}(\mathbf{x};\boldsymbol{\mu}))} \left(\left.\frac{-\mathrm{d}f_T(\mathbf{t})}{\mathrm{d}\mathbf{t}}\right|_{\mathbf{t}=\mathbf{g}(\mathbf{x};\boldsymbol{\mu})}\right)^{\mathsf{T}} \frac{\partial \mathbf{g}(\mathbf{x};\boldsymbol{\mu})}{\partial \mu}. \tag{16}$$

4.3. Hessian

Let us define the matrix of the second derivative of S as its Hessian $\Delta^2 S$

$$\Delta^2 S = \begin{bmatrix} \frac{\partial^2 S}{\partial \mu_1 \partial \mu_1} & \frac{\partial^2 S}{\partial \mu_1 \partial \mu_2} & \cdots \\ \frac{\partial^2 S}{\partial \mu_2 \partial \mu_1} & \frac{\partial^2 S}{\partial \mu_2 \partial \mu_2} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}. \tag{17}$$

With the same assumptions as before, we determine a component of the Hessian by

$$\begin{aligned} \frac{\partial^2 S}{\partial \mu_1 \partial \mu_2} \quad = \quad & -\left(\sum_{\iota \in L_T} \sum_{\kappa \in L_R} \frac{\partial^2 p(\iota, \kappa)}{\partial \mu_1 \partial \mu_2} \log_2 \frac{p(\iota, \kappa)}{p_T(\iota)}\right) + \\ & \frac{1}{\log_e 2} \left(\sum_{\iota \in L_T} \frac{\partial p_T(\iota)}{\partial \mu_1} \frac{\partial p_T(\iota)}{\partial \mu_2} \frac{1}{p_T(\iota)}\right) - \frac{1}{\log_e 2} \left(\sum_{\iota \in L_T} \sum_{\kappa \in L_R} \frac{\partial p(\iota, \kappa)}{\partial \mu_1} \frac{\partial p(\iota, \kappa)}{\partial \mu_2} \frac{1}{p(\iota, \kappa)}\right). \end{aligned} \tag{18}$$

The first term of (18) depends on the second-order variation of the joint probability when a pair of registration parameters varies jointly. We will ignore this term, which amounts to linearizing the variation of p with respect to $\boldsymbol{\mu}$. Another motivation for dropping the first term in (18) arises when one considers the situation at ideal registration. In this case, the parameters $\boldsymbol{\mu}$ are such that the dependence between the transformed test image $f_T(\mathbf{g}(\mathbf{x}))$ and the reference image $f_R(\mathbf{x})$ is complete; that is, there exists an exact mapping $\kappa_0(\iota)$ such that $p(\iota, \kappa_0(\iota)) = p_T(\iota) = p_R(\kappa_0(\iota))$. The logarithmic term vanishes for such cases, and there is no contribution to the first term of $\partial^2 S / \partial\mu_1\, \partial\mu_2$ for all entries (ι, κ) that satisfy this mapping. In addition, there is no contribution to the gradient $\boldsymbol{\Delta} S$ either. All other entries $\kappa \neq \kappa_0(\iota)$ in $p(\iota, \kappa)$ are zero, which we have shown not to contribute to the criterion S. Finally, at ideal registration, not only do we have $\boldsymbol{\Delta} S = \mathbf{0}$, which was expected, but also the first term in (18) disappears. Thus, in this paper we use the following simplified form

$$\frac{\partial^2 S}{\partial\mu_1\, \partial\mu_2} \approx \frac{1}{\log_e 2} \left(\sum_{\iota \in L_T} \frac{\partial p_T(\iota)}{\partial\mu_1} \frac{\partial p_T(\iota)}{\partial\mu_2} \frac{1}{p_T(\iota)} \right) - \frac{1}{\log_e 2} \left(\sum_{\iota \in L_T} \sum_{\kappa \in L_R} \frac{\partial p(\iota,\kappa)}{\partial\mu_1} \frac{\partial p(\iota,\kappa)}{\partial\mu_2} \frac{1}{p(\iota,\kappa)} \right). \tag{19}$$

Comparing this last expression with (13) and (16), one sees that every term needed by our simplified Hessian has been already precomputed while determining the value of the gradient. Thus, another fortunate consequence of ignoring the second-order term in (18) is that the Hessian $\Delta^2 S$ comes at essentially no additional computational cost with respect to the gradient $\boldsymbol{\Delta} S$.

4.4. Standard optimizers

The steepest-gradient descent is a minimization algorithm that can be succinctly described by

$$\boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} - \Gamma\, \boldsymbol{\Delta} S(\boldsymbol{\mu}^{(k)}). \tag{20}$$

Its convergence is guaranteed, although it may be very slow. A key problem is the determination of the appropriate scaling diagonal matrix Γ.

The Newton method can be described by

$$\boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} - \left(\Delta^2 S(\boldsymbol{\mu}^{(k)}) \right)^{-1} \boldsymbol{\Delta} S(\boldsymbol{\mu}^{(k)}). \tag{21}$$

Its convergence to an optimum is not guaranteed: it may converge to a saddle point (at the same time a maximum for some parameter μ_1 and a minimum for another parameter μ_2). Even worse, it diverges from the desired solution when the problem is not convex. In return, it is extremely efficient when the criterion is locally quadratic, for in this case it finds the optimum in a single iteration.

4.5. Marquardt-Levenberg strategy

The Marquardt-Levenberg strategy is a convenient way to combine the advantages of the gradient method with those of the Newton method, preserving the efficiency of the latter when the conditions are nearly optimal, and the robustness of the former when they are not. Trying first to alleviate the scaling problem for Γ, let us examine the dimensions $\langle \bullet \rangle$ of each component involved in (20). We conclude that a component μ_i needs an individual factor γ_i of dimension

$$\langle \gamma_i \rangle = \langle \mu_i \rangle^2 \langle S \rangle^{-1}, \tag{22}$$

which is exactly the inverse of the corresponding main diagonal entry of the Hessian matrix $[\Delta^2 S]_{i,i}$, in terms of dimensional units. Selecting the latter as γ_i still does not necessarily provide any really appropriate value, but at least it offers a way to impose an order of magnitude to the ratio of the weights associated with each component i. We correct this value by introducing a global, component-independent tuning factor λ, which at the same time allows one to recover the adaptation mechanism typical of a gradient approach, and to correct for potential inadequacies in the values $1/[\Delta^2 S]_{i,i}$. For a component i, the gradient approach becomes

$$\mu_i^{(k+1)} = \mu_i^{(k)} - \frac{1}{\lambda} \frac{1}{[\Delta^2 S(\boldsymbol{\mu}^{(k)})]_{i,i}} [\boldsymbol{\Delta} S(\boldsymbol{\mu}^{(k)})]_i. \tag{23}$$

Let us introduce a modified Hessian $\mathcal{H}S$ in which we retain the off-diagonal entries of $\Delta^2 S$ and multiply its diagonal entries by some factor

$$[\mathcal{H}S(\boldsymbol{\mu})]_{i,j} = [\Delta^2 S(\boldsymbol{\mu})]_{i,j}\,(1 + \delta_{i,j}\,\lambda), \tag{24}$$

where $\delta_{i,j}$ is the Kronecker symbol. Suppose we now determine the new update $\boldsymbol{\mu}^{(k+1)}$ as in

$$\boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} - (\mathcal{H}S(\boldsymbol{\mu}^{(k)}))^{-1}\,\boldsymbol{\Delta}S(\boldsymbol{\mu}^{(k)}). \tag{25}$$

Depending on the value of λ, one can distinguish two extreme cases. On one hand, when $\lambda \to 0$, one sees that (25) and (21) are identical. On the other hand, when $\lambda \to +\infty$, the diagonal terms of the modified Hessian $\mathcal{H}S$ dominate, and we are in the situation of (23). Note however that, although the magnitude of the update is adapted to each component by the virtue of the normalizing term $[\Delta^2 S]^{-1}_{i,i}$, the steps are vanishingly small. This should not be a problem, because it is easy to establish λ between these two extremes in order to achieve a good compromise between the efficiency (but lack of robustness) of the Newton approach, and the size of the steps of the robust (but generally inefficient) gradient approach.

The way to perform this adaptation is as follows: given an initial $\lambda^{(k)}$ and a parameter $\boldsymbol{\mu}^{(k)}$, determine a new parameter $\boldsymbol{\mu}_0^{(k+1)}$ according to (25). Compare $S(\boldsymbol{\mu}^{(k)})$ with $S(\boldsymbol{\mu}_0^{(k+1)})$. If there is an improvement, we are nearing the solution and a Newton approach is better suited. In this case, reduce the value of $\lambda^{(k)}$ by some arbitrary factor $\alpha' > 1$ such that $\lambda_0^{(k+1)} = \lambda^{(k)}/\alpha'$. Start the procedure again with $\lambda_0^{(k+1)}$ in the role of $\lambda^{(k)}$ and $\boldsymbol{\mu}_0^{(k+1)}$ in the role of $\boldsymbol{\mu}^{(k)}$. Otherwise, when $\boldsymbol{\mu}_0^{(k+1)}$ fails to improve the criterion, it is likely that we are not sufficiently close to the solution, and the truncated Taylor expansion we considered is not a good approximation of its full expansion (10). Note that the simplification of the Hessian given by (19) and its further refinement given by (24) might also be blamed. In this situation, a gradient approach is indicated. Starting with $\lambda_0^{(k+1)} = \lambda^{(k)}$, magnify $\lambda_0^{(k+1)}$ by some arbitrary factor $\alpha'' > 1$ such that $\lambda_{n+1}^{(k+1)} = \alpha''\,\lambda_n^{(k+1)}$. Use (25) with increasing n to determine potential candidates $\boldsymbol{\mu}_{n+1}^{(k+1)}$, until some N is found for which the criterion improves. Then, start the procedure again with $\lambda_N^{(k+1)}$ in the role of $\lambda^{(k)}$ and $\boldsymbol{\mu}_N^{(k+1)}$ in the role of $\boldsymbol{\mu}^{(k)}$.

5. EXPERIMENTS

We want to illustrate the performance of our approach with experiments using a pair of biomedical images coming from different modalities. Figure 1 shows such a pair, where the left image is the red channel of the cryosection of a human brain (Slice 4125 of the Visible Human Project), and the right image is its blue channel. Since they come from the same 24 bit color photograph, we can have some confidence in their correct prior alignment, up to the scanner accuracy. In addition, we have reduced their size threefold to 256×256, which tends to reduce any original mismatch in the color channels by as much. The general procedure for validating our algorithm will be to impose a transformation on one or on both of the images, and to try to recover its inverse by our registration method.

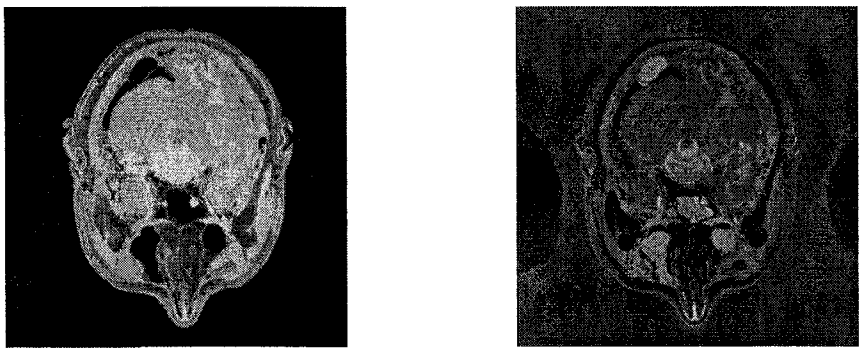

Figure 1. Cryosection of a human brain in an RGB representation. Left: red channel. Right: blue channel.

5.1. Warping index

The test image f_T and the reference image f_R are supposed to be already in perfect correspondence. Rather than transforming a single image in this pair, we prefer to transform both because this tends to limit any bias, by distributing equally the artifacts introduced by the interpolation procedure. Therefore, we compute

$$g_R(\mathbf{x}) = f_R(\mathbf{g}_0(\mathbf{x})),\ g_T(\mathbf{x}) = f_T(\mathbf{g}_0^{-1}(\mathbf{x})), \tag{26}$$

where $\mathbf{g}_0$ is a rigid-body transformation consisting of a random translation and a random rotation around the center of the image. It follows that the correct registration of g_T to g_R involves the transformation $\mathbf{g} = \mathbf{g}_0 \circ \mathbf{g}_0$ such that $g_R(\mathbf{x}) = g_T(\mathbf{g}(\mathbf{x}))$.

Next, we apply any of several registration methods to estimate a transformation $\tilde{\mathbf{g}}$ out of the data (g_T, g_R). Our aim is now to determine the precision of each estimation. We achieve this goal by introducing a warping index ϖ that measures an average geometric error

$$\varpi = \frac{1}{(\#V)} \sum_{\mathbf{x} \in V} \|\mathbf{g}^{-1}(\mathbf{x}) - \tilde{\mathbf{g}}^{-1}(\mathbf{x})\|, \tag{27}$$

where $\|\bullet\|$ stands for the Euclidean distance. After having performed several registrations with different realizations of the random transformation $\mathbf{g}_0$, we average together the values ϖ and report a pooled warping index. For this paper, there are 50 warping indexes to pool for each experiment. Meanwhile, $\mathbf{g}_0$ has a translation that is uniformly distributed in $[-2.5, 2.5]$, and a rotation around the center of the image that is uniformly distributed in $[-\pi/36, \pi/36]$. Hence, the maximal excursion of $\mathbf{g}$ is about 7 pixels of translation and 10° of rotation.

5.2. Multiresolution

We expect a multiresolution approach to influence at least two aspects of the registration method. First, it should improve the robustness of non-stochastic optimization procedures such as ours. Second, it should improve its speed. To observe these effects, we compare the success of our registration method when we vary the number of levels in the image pyramid. Since the goal is to investigate robustness, we prescribe a fixed overall computation time and adapt the number of iterations at each level such that this resource is shared between levels in an adequate fashion.

Table 1 presents the results of these experiments where the first line corresponds to a strategy using a one-level 'pyramid', and the last line to a four-level pyramid. The geometric unit is one pixel at the finest resolution, and the time unit is one CPU second on a SPARC 20 workstation. Out of 50 trials, we retain in this table only those for which the quality of the registration is at least subpixel; we consider the rejected cases to be failures. We observe that our algorithm is essentially unable to converge within the allotted computation time when the pyramid consists of its finest level only. With two levels, the number of failures is reduced but is still significant. Also, both accuracy and capture range are smaller than for the three-level case, where we experience no failures at all and where the order of magnitude of accuracy is only a few hundredths of a pixel. These results are left unchanged by the introduction of an additional fourth level, although the standard deviation of ϖ is further reduced*.

One would believe that the many failures observed when the image pyramid holds but a single level are solely due to an insufficient computation time. This is not true. Table 2 shows the result of a tenfold increase in the number of iterations. Even in this case, failures are more numerous than successful registrations. This happens because the criterion $S(\mu)$ is very detailed at the finest resolution, which leaves many opportunities for a non-stochastic optimization algorithm to get trapped into a local optimum. When the image resolution gets coarser, the amount of detail decreases and local optima tend to disappear.

5.3. Quality of the model

We expect the quality of the image model to reflect itself in the quality of registration, particularly at the coarse levels of the pyramid. To investigate this hypothesis, we construct Table 3 where we show the results of registration using cubic, quadratic and linear models, respectively. The number of levels and the number of iterations are identical in these three cases.

*The residual error might also be in part due to inaccuracies in the scanning device itself.

Table 1. Influence of multiresolution on the robustness of registration.

	Initial	Coarse	...	...	Fine	Total time	Failures
ϖ	1.32				0.91		
Time/Iterations					95.8/11	95.8	98%
ϖ	2.43 ± 0.59			0.24 ± 0.29	0.16 ± 0.24		
Time/Iterations				55.4/24	32.1/4	87.5	66%
ϖ	4.57 ± 1.99		0.12 ± 0.07	0.04 ± 0.02	0.04 ± 0.01		
Time/Iterations			38.8/64	19.1/8	32.1/4	90.0	None
ϖ	4.57 ± 1.99	1.41 ± 0.83	0.12 ± 0.07	0.04 ± 0.02	0.04 ± 0.01		
Time/Iterations		12.2/64	20.7/32	19.1/8	32.1/4	84.1	None

Table 2. Registration without multiresolution.

	Initial	Fine	Total time	Failures
ϖ	2.58 ± 0.74	0.09 ± 0.17		
Time/Iterations		914.1/100	914.1	60%

The quality of the model affects both interpolation and pyramid computation. One can see that the difference between a cubic and a quadratic model is not striking when dealing with the finer levels of the pyramid. For the coarser levels however, the difference is more marked. This tends to show that the main advantage of using a cubic model (with respect to a quadratic one) is not so much due to interpolation, but rather to reduced aliasing in the pyramid. Note that quadratic and cubic models have essentially the same computational cost, while a linear model is somewhat cheaper. The gain is not dramatic however, and has to be weighted against a sharp reduction in accuracy. Moreover, since the algorithm sometimes failed to converge with a linear model, robustness is also decreased. For all these reasons we advocate the use of a cubic model.

5.4. Optimizer

We want now to compare the accuracy and efficiency of our proposed optimizer to the Powell algorithm that has been also used in the context of image registration based on mutual information.[11] The Powell algorithm uses only function evaluations while attempting to recover the gradient $\mathbf{\Delta} S$ and the Hessian $\Delta^2 S$. Thus, it foregoes the computation of explicit derivatives, which makes it an attractive candidate when closed forms are not available or when their computation cost is prohibitive. It is known as a direction-set method, for which the parameter space is explored along straight lines exclusively (linear combinations of $(\mu_1, \mu_2, \ldots)$). Unfortunately, line optimization processes are not very efficient, for they first require a bracketing of the optimum along the considered line before being able to start the optimization itself. This bracketing alone costs several function estimations. Obviously, the fact that the Powell algorithm uses estimates for the derivatives rather than their true values tends also to further reduce its efficiency. Moreover, it is sensitive to the order in which the first few line optimizations are performed.

We conduct an experiment where the transformation $\mathbf{g}$ corresponds to an initial displacement of about 5 pixels along each axis and to a rotation of $\pi/18 = 10°$. Using a four-level pyramid, we observe the evolution of the warping index ϖ during the course of registration. The image model is cubic for both algorithms. Figure 2 shows the result of this experiment, where the change of pyramid level occurred after having performed 256, 128, 64 and 32 iterations at each level, respectively. The bracketing episodes of the Powell algorithm can be easily identified as big excursions of ϖ. Those are necessary because this algorithm has no indication of the correct scale of the optimization problem and has to start with wild guesses each time a new direction is tried. The reward is a reduction in complexity, since no explicit derivative computations are performed. This translates in a reduction of the time per iteration by a factor two with respect to our algorithm.

Figure 2 shows that our optimizer converges generally in much fewer iterations than Powell, but for the coarsest resolution. This was expected, because our algorithm reaps most of its benefits only when the starting conditions are

Table 3. Influence of the model degree on the robustness of registration.

	Initial	Coarse	...	...	Fine	Total time	Failures
$\varpi(\beta^{(3)})$	4.57 ± 1.99	1.41 ± 0.83	0.12 ± 0.07	0.04 ± 0.02	0.04 ± 0.01		
Time/Iterations		12.2/64	20.7/32	19.1/8	32.1/4	84.1	None
$\varpi(\beta^{(2)})$	4.57 ± 1.99	1.57 ± 0.96	0.13 ± 0.08	0.04 ± 0.02	0.04 ± 0.01		
Time/Iterations		11.8/64	20.5/32	19.0/8	32.6/4	83.9	None
$\varpi(\beta^{(1)})$	4.31 ± 1.89	3.47 ± 1.46	0.52 ± 0.42	0.16 ± 0.22	0.11 ± 0.17		
Time/Iterations		8.4/64	15.1/32	14.1/8	24.7/4	62.3	10%

nearly optimal, which they are not at the very first level. Since we spend twice as much time per iteration, the time to convergence at the coarsest level is twice as long. This is not critical, because the computational burden at coarse levels, even with many iterations, is small compared to the cost of just a few iterations at higher levels. When it is indeed time to switch to the next level, our algorithm converges in very few iterations, while Powell has to wait until one or another of its line optimizations is performed along a useful direction in the space of parameters[†]. Altogether, and notwithstanding the longer time needed by our algorithm for performing any single iteration, we converge much earlier than the Powell algorithm.

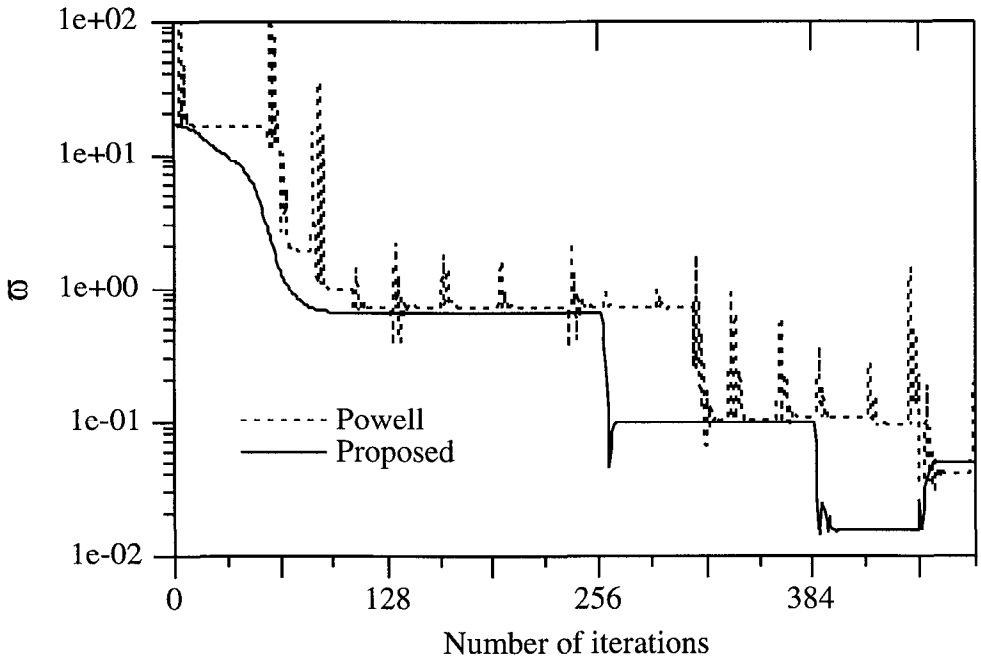

Figure 2. Comparison of the accuracy of our proposed algorithm and the accuracy of the Powell algorithm.

We also take advantage of this experiment to show the relationship between the measure of geometric accuracy ϖ and the value taken by the criterion S during the course of optimization by our algorithm. It can be seen in Figure 3 that this relationship is very close to being monotonic, with some very small departure near the hypothetic optimum. We have indicated by a cross the optimal S found at each resolution level. There are two likely explanations why

[†] It is sometimes possible to tune the Powell algorithm in such a way that the most promising directions are tried first, and to adapt to a specific problem the mechanism that decides to abandon the current direction for the exploration of a new one. However, this tuning depends heavily on the application.

the best result, in terms of ϖ, has not been selected by the algorithm. Either it is unable to reach the true optimum, or the hypothesis according to which $\varpi = 0$ corresponds to a perfect registration is slightly wrong. To test these issues, we have repeated the experiment presented at Table 3 with $\beta^{(3)}$, while using only the red channel of Figure 1 in the role of both the test and reference image. In this case, a perfect registration corresponds necessarily to $\varpi = 0$, and we observe that the resulting pooled error is halved. This might indicate that, in the previous experiments, a slightly imprecise alignment of the color channels might be responsible for some fraction of ϖ. This may also be the explanation for the apparent increase of the warping index during the very final stage of the optimization presented in Figure 2.

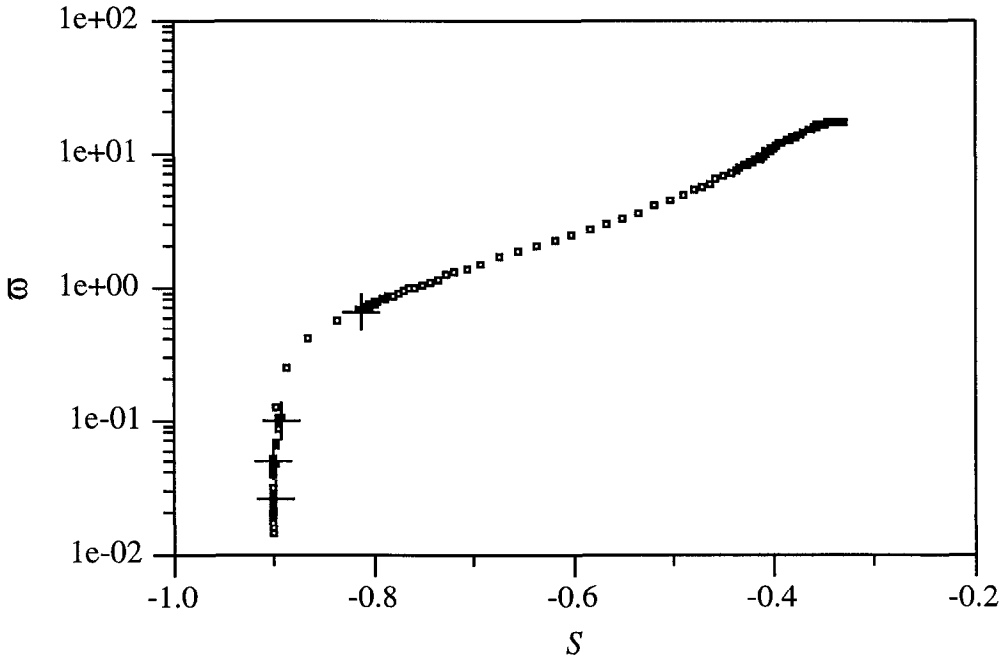

Figure 3. Dependence of the geometric distortion (or warping) ϖ on the criterion S.

6. CONCLUSIONS

We have developed a new optimizer for solving the problem of inter-modal image registration. This optimizer takes benefit of the Marquardt-Levenberg strategy, while extending its capabilities to a specific problem that does not involve a least-squares criterion. The optimized criterion is the mutual information between the two images to register. We propose to compute its value through the use of separable Parzen window. We show that the selection of a Parzen window that satisfies the partition of unity simplifies several aspects of the problem. It allows us to find a tractable closed form expression for the gradient of the criterion with respect to the transformation parameters, and to justify a simplified form for its Hessian as well. Moreover, the partition of unity guarantees that the histogram of the fixed reference image does not depend on the transformation. We have introduced a coherent framework based on a continuous image model for applying the transformations and computing the derivatives of the criterion. The same model is used for performing the registration in a multiresolution context. Both model and Parzen window are based on B-splines. We have shown experimentally that our new optimizer is well adapted to multiresolution processing, which brings robustness and speed to the whole approach. We reach a better accuracy in less time than previous published methods.

Our registration method has been applied with good success to the registration of volume brain images across several modalities (CT, MRI, PET, T1, T2). These results will be presented in a forthcoming paper.

ACKNOWLEDGEMENTS

We wish to thank Murray Eden for his kind help in revising our manuscript.

REFERENCES

1. L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys* **24**(4), pp. 325–376, 1992.
2. P. A. Van den Elsen, E.-J. D. Pol, and M. A. Viergever, "Medical image matching—a review with classification," *IEEE Engineering in Medicine and Biology Magazine* **12**(1), pp. 26–39, 1993.
3. A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal, *Automated Multi-Modality Image Registration Based on Information Theory*, pp. 263–274. Kluwer Academic, 1995.
4. P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *Proc. International Conference on Computer Vision* , pp. 16–23, (Boston, MA, USA), June 1995.
5. A. Collignon, D. Vandermeulen, P. Suetens, and G. Marchal, "3D multi-modality medical image registration using feature space clustering," *Proc. Computer Vision, Virtual Reality, and Robotics in Medicine* , pp. 195–204, (Nice, France), April 1995.
6. M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I—theory," *IEEE Transactions on Signal Processing* **41**(2), pp. 821–832, 1993.
7. M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part II—efficient design and applications," *IEEE Transactions on Signal Processing* **41**(2), pp. 834–848, 1993.
8. M. Unser, A. Aldroubi, and M. Eden, "The L2 polynomial spline pyramid," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**(4), pp. 364–379, 1993.
9. A. Aldroubi and M. Unser, "Sampling procedures in function spaces and asymptotic equivalence with Shannon's sampling theorem," *Numerical Function Analysis and Optimization* **15**(1&2), pp. 1–21, 1994.
10. D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics* **11**(2), pp. 431–441, 1963.
11. F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE Transactions on Medical Imaging* **16**(2), pp. 187–198, 1997.