Fast Geodesic Active Fields - A Splitting Scheme for Geometric Image Registration

Dominique Zosso*, *Student Member, IEEE,* Xavier Bresson, Jean-Philippe Thiran, *Senior Member, IEEE*

***Abstract*—In this paper we present an efficient numerical scheme for the recently introduced Geodesic Active Fields (GAF) framework for geometric image registration. This framework considers the registration task as a weighted minimal surface problem. Hence, data-term and regularization term are combined trough multiplication in a single, parametrization invariant and geometric cost functional. The multiplicative coupling provides an intrinsic spatially varying and data-dependent tuning of the regularization strength, while the parametrization invariance allows working with images of non-flat geometry, generally defined on any smoothly parametrizable manifold. The directly resulting minimizing energy flow, however, has poor numerical properties. Here, we provide an efficient numerical scheme that uses a splitting approach: data and regularity term are optimized over two distinct deformation fields, that are constrained to be equal. Our approach is more flexible than standard Gaussian regularization, since one can interpolate freely between isotropic Gaussian and anisotropic TV-like smoothing. In this work, we compare the Geodesic Active Fields method against the popular and fast state-of-the-art Demons method. Overall, we can show the advantages of the proposed FastGAF method. It compares favorably against Demons, both in terms of registration speed and quality.**

***Index Terms*—Augmented Lagrangian, Biomedical image processing, Computational geometry, Diffusion equations, Geodesic Active Fields, Image registration, Non-convex optimization.**

I. Introduction

IMAGE REGISTRATION is the concept of mapping homologous points of different images, representing a same object. In practice, however, it is often difficult to establish homology in images based on this definition. For automatic image registration, it is commonplace to substitute homology by a measurable criterion of image dissimilarity, which is to be minimized by an unknown deformation field $\mathbf{u}$. The determination of this deformation field is an ill-posed inverse problem, and regularity constraints are usually introduced to solve it.

A multitude of approaches to solve the registration problem exist, see [1]–[4] and references therein. Here, we focus on the non-parametric variational approach, where image distance metric and regularization penalty are commonly incorporated into a single energy minimization model, a.k.a. variational model, e.g., [5]. The energy functionals are commonly of the general form

$$E = E_{\text{data}} + \gamma \cdot E_{\text{regularization}}, \quad \gamma > 0. \tag{1}$$

We provide a more detailed description of the very popular Demons algorithm [6], since it will serve as a baseline method in this study. The fundamental Demons energy is given by

$$\begin{aligned} E_{Demons} &= \|\mathcal{M}(\mathbf{u}) - \mathcal{F}\|^2 + \gamma\|\nabla\mathbf{u}\|^2 \\ &= \int_\Omega |\mathcal{M}(\mathbf{x}+\mathbf{u}(\mathbf{x})) - \mathcal{F}(\mathbf{x})|^2 \,\mathrm{d}\mathbf{x} + \gamma\int_\Omega |\nabla\mathbf{u}(\mathbf{x})|^2 \,\mathrm{d}\mathbf{x} \end{aligned} \tag{2}$$

where $\mathcal{M},\mathcal{F} : \mathbf{x} \in \Omega \subset \mathbb{R}^n \to \mathbb{R}_+$ are respectively the moving and fixed image, and $\mathbf{u} : \mathbf{x} \in \Omega \subset \mathbb{R}^n \to \mathbb{R}^p$ is the deformation field with p components. This corresponds to a high-pass Thikhonov-regularized minimization of the L_2-norm of the image-difference [7], [8]. A recent and rather efficient scheme makes use of a hidden variable, the so called correspondences [9]. In the end, this amounts to an operator splitting scheme, where additive minimization of image distances on the one deformation field, and regularization energy on the other, hidden, deformation field, are coupled through an ordinary penalty term:

$$E_{Demons} = \frac{1}{\sigma_i^2}\|\mathcal{M}(\mathbf{u}) - \mathcal{F}\|^2 + \frac{1}{\sigma_x^2}\|\mathbf{u}-\mathbf{v}\|^2 + \frac{1}{\sigma_T^2}\|\nabla\mathbf{v}\|^2, \tag{3}$$

where σ_i, σ_x and σ_T are balancing parameters that correspond to the noise on the image intensities, the uncertainty of spatial correspondences, and to the amount of regularization that is desired, respectively.

In its standard variant, the above Demons energy is minimized in a two step iterative process:

$$\mathbf{u}^{k+1} = \arg\min_{\mathbf{u}} \left\{ \frac{1}{\sigma_i^2}\|\mathcal{M}(\mathbf{u}) - \mathcal{F}\|^2 + \frac{1}{\sigma_x^2}\|\mathbf{u}-\mathbf{v}^k\|^2 \right\} \tag{4}$$

$$\mathbf{v}^{k+1} = \arg\min_{\mathbf{v}} \left\{ \frac{1}{\sigma_T^2}\|\nabla\mathbf{v}\|^2 + \frac{1}{\sigma_x^2}\|\mathbf{u}^{k+1}-\mathbf{v}\|^2 \right\}. \tag{5}$$

Given an L_2 norm on the image difference, then the first minimization step is a non-linear least squares problem, and it can be solved using the Gauss-Newton method. The second minimization task is particularly simple, and it is solved through convolution with a Gaussian-like smoothing kernel, often even agnostic, i.e. the Gaussian kernel is tuned without actually deriving it from the energy term [9]. A more detailed derivation of the convolution kernel as energy minimizer is given in the appendix.

Demons are today a commonly used method in medical image registration because of their rapidity. However, Gaussian convolution is only fast on Cartesian grids, and further the

This work is supported by the Swiss National Competence Center in Biomedical Imaging (NCCBI). D. Zosso and J.-P. Thiran are with the Signal Processing Laboratory (LTS5), Ecole Polytechnique Fédérale de Lausanne (EPFL), Station 11, CH-1015 Lausanne, Switzerland, {dominique.zosso,jp.thiran}@epfl.ch. X. Bresson is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong SAR, xbresson@cityu.edu.hk

linear diffusion regularization might not be suitable for all registration tasks.

Therefore, in previous work [10], we propose to represent the deformation field as a hypersurface embedded in a higher dimensional space, inspired by the Beltrami framework [11]. The energy of the hypersurface belongs to the class of (weighted) minimal surface problems. In other words, the minimization flow drives the deformation field towards a harmonic map corresponding to the solution of the registration problem. The proposed energy is measured with the Polyakov energy [12], weighted by a suitable image distance borrowed from standard registration models.

This registration method shares important properties and similarities with the well-known geodesic active contours (GAC) model [13] in image segmentation. For example, our registration method is re-parametrization invariant, like the GAC model. Further, our registration method can be designed to work on any kind of surface, such as the sphere. In [10] we have introduced the concept of GAF along with some prototype results for purely illustrative purposes. There, we focused explicitly on the theoretical aspects of GAF and its conceptual strengths, e.g. re-parametrization invariance and applications on non-Euclidean images. The prototype implementations served as a proof of concept, while their main drawback was the relatively poor computational performance, compared to state-of-the-art methods, including Demons.

Here, we want to address these numerical and performance shortcomings and provide a numerical scheme that considerably improves the speed of the GAF energy minimization. The basic idea is to use a splitting method borrowed from optimization theory, such as [14], [15], which aims at solving some optimization problems efficiently through simpler sub-optimization problems. In our case, it consists in splitting the data term and the regularization term, which are coupled with the product operator. The splitting is processed in an augmented Lagrangian approach to guarantee as much as possible the equivalence with the original optimization problem [16]. However, we observe that the splitting method is applied to a non-convex functional (product of the data term and the regularization term) which is thus not necessarily guaranteed to converge to the optimal solution, even if all our experiments converged to satisfying solutions. The main interest of the proposed method is the splitting of the rather complex GAF problem into smaller sub-problems, for which fast resolution schemes exist. While we describe promising solvers for those sub-problems, note that those can still be improved, which is, however, beyond the scope of this paper.

The rest of this paper is organized as follows. First, we will briefly recall the GAF framework in section II, and summarize a direct implementation. Then, in section III we provide a short introduction into the method of augmented Lagrangians (AL) for optimization with constraints. In section IV we will make use of a splitting scheme and AL to get a fast minimization scheme for the GAF energy. We setup both, a stereo vision and a full 2D registration test case, in section V. Results will be presented in section VI, and we will give a short conclusion and outlook in section VII.

II. Geodesic Active Fields

In this section we recall the GAF framework for image registration, introduced in [10]. The deformation field is embedded as a mapping between the n-dimensional image domain and a m-dimensional space, where $m > n$. This is achieved by letting the components of the deformation field become additional dimensions of the embedding space, in close analogy to the Beltrami framework for image smoothing and restoration [11]. The embedded manifold then evolves towards a weighted minimal surface, while being attracted by a deformation field that brings the two images into registration. The main strengths of this framework are:

1) The freedom to register images on any Riemannian manifold, i.e., on any smooth and parametrized surface.
2) The invariance under re-parametrization of the proposed energy, like the GAC energy [13] for the segmentation problem.
3) The data-dependent adaptation of the regularization strength thanks to the multiplicative weighting.
4) The ability to work with multi-scale images, where the relation between space and scale is inherently accounted for.

A. The GAF framework

In the most general form, we register a pair of n-dimensional images defined on a Riemannian domain $\Omega \subset \mathbb{R}^n$ with coordinates $\mathbf{x} = (x_1, \dots, x_n)$. The deformation field acts along $p \leq n$ dimensions, i.e., $\mathbf{u} : \Omega \mapsto \mathbb{R}^p, \mathbf{u}(\mathbf{x}) = (u_1(\mathbf{x}), \dots, u_p(\mathbf{x}))$. At the very core of GAF, the deformation field is seen as a surface or hypersurface embedded in a higher dimensional space, much like images embedded with the Beltrami framework [11]. On these embeddings, a Riemannian structure can be introduced: the metric $g_{\mu\nu}$ locally measures the distances on the embedded deformation field, whereas in the higher dimensional embedding space distances are measured using h_{ij}.

The embedding $X \in \mathbb{R}^m$, $m = n+p$, and the metric tensors $h_{ij} \in \mathbb{R}_+$, $i, j \in [1, m]$ and $g_{\mu\nu} \in \mathbb{R}_+$, $\mu, \nu \in [1, n]$ are chosen as follows:

$$\begin{cases} X : (x_1, \dots, x_n) \to (x_1, \dots, x_n, u_1, \dots, u_p) \\ h_{ij} \text{ is arbitrary} \\ g_{\mu\nu} = \partial_\mu X^i \partial_\nu X^j h_{ij}, \end{cases} \tag{6}$$

where $x_1, \dots, x_n$ denote the spatial components of the image and $u_1, \dots, u_p$ are the components of the dense deformation field. Based on this choice, we define the following general registration energy functional, which is a weighted Polyakov energy [12], [17], for the geodesic active fields (GAF):

$$E_{GAF} = \int f \sqrt{g} \, \mathrm{d}\mathbf{x} \tag{7}$$

where the weighting function $f = f(\mathbf{x}, \mathbf{u})$ is arbitrary, and represents a penalty with respect to the alignment of the two images, as detailed below. Now, considering the PDE

$$\partial_t X^i = -\frac{1}{2\sqrt{g}} h^{il} \frac{\partial E_{GAF}}{\partial X^l}, \tag{8}$$

we get the following minimizing flow on the deformation field:

$$\partial_t u_i = fH^{n+i} + \partial_k f g^{\mu\nu} \partial_\mu X^k \partial_\nu u_i - \frac{m \cdot n}{2} \partial_k f h^{k(n+i)}, \quad 1 \leq i \leq p \tag{9}$$

where H^{n+i} stands for the component of the manifold mean curvature vector corresponding to the deformation field dimension i. This is the Laplace-Beltrami operator

$$H^{n+i} = \frac{1}{\sqrt{g}} \partial_\mu \sqrt{g} g^{\mu\nu} \partial_\nu u_i =: \Delta_g u_i. \tag{10}$$

B. Weighting function

The purpose of the weighting function f is to drive the deformation field toward minimal surfaces that bring the two images into registration. As such, the flow must stop when the deformed image perfectly matches the target image. Hence, the weighting function is naturally chosen to be an image distance metric, which approaches zero when the two images locally match. An intuitive primer for monomodal image registration is the squared error metric [18], leading to:

$$f(\mathbf{x}, \mathbf{u}) = 1 + \alpha \cdot \left(\mathcal{M}(\mathbf{x} + \mathbf{u}) - \mathcal{F}(\mathbf{x}) \right)^2, \tag{11}$$

where $\mathcal{F}$ and $\mathcal{M}$ refer to the fix and moving images, respectively. In other cases, e.g., for stereo vision, the absolute error norm might be more appropriate:

$$f(\mathbf{x}, \mathbf{u}) = 1 + \alpha \cdot \left| \mathcal{M}(\mathbf{x} + \mathbf{u}) - \mathcal{F}(\mathbf{x}) \right|, \tag{12}$$

For other examples, e.g., suitable for multimodal image registration, or more complicated deformation models, refer to [10], [19].

C. The Stereo vision case

Let us consider a case of stereo vision depth recovery of 2D images [20]. This choice works on 2D images, $n = 2$, and reduces the co-dimension of the deformation field to $p = 1$, which heavily simplifies the notation. The schemes, however, generalize directly to higher co-dimensions, e.g., for full 2D image registration, and can be adapted to other image geometries as well. The different components of the deformation field optimize separately, the only time they are considered jointly is to compute the metric tensor $g_{\mu\nu}$ and the weighting function.

The stereo embedding is defined as in [10]:

$$\begin{cases} X : (x, y) \to (x, y, u) \\ h_{ij} = \mathrm{diag}(1, 1, \beta^2) \\ g_{\mu\nu} = \begin{bmatrix} 1 + \beta^2 u_x^2 & \beta^2 u_x u_y \\ \beta^2 u_x u_y & 1 + \beta^2 u_y^2 \end{bmatrix} \\ g = 1 + \beta^2 |\nabla u|^2. \end{cases} \tag{13}$$

Put into the general equations we get the following energy functional:

$$\begin{cases} E_{GAF} = \int f \sqrt{1 + \beta^2 |\nabla u|^2} \mathrm{d}x \mathrm{d}y \\ \partial_t u = fH^u + \partial_k f g^{\mu\nu} \partial_\mu X^k \partial_\nu u - \frac{3}{\beta^2} f_u. \end{cases} \tag{14}$$

D. Discretization

As we work with sampled images and deformation fields, we adapt the notation of variables and the energy accordingly. First, let $\mathbf{x} \in \mathbb{R}^{N \times n}$ denote the matrix containing the coordinates of all samples, where N is the number of spatial samples, and where n is the dimension of the images. Pixel i is located at $\mathbf{x}_i \in \mathbb{R}^n$. Similarily, $\mathbf{u}_i = \mathbf{u}(\mathbf{x}_i) \in \mathbb{R}^p$ is the deformation vector at $\mathbf{x}_i$. For any $i \in [1, N]$ let us write $x_{i,j}$, $j \in [1, n]$ and $u_{i,q}$, $q \in [1, p]$ to denote a specific component of those vectors. Finally, the weighting function and the square root of the metric tensor are rewritten as column vectors, $F, G \in \mathbb{R}^N$, respectively:

$$F_i = f(\mathbf{x}_i, \mathbf{u}_i) \ \ \forall i \in [1, N] \tag{15}$$

$$G_i = \sqrt{g(\mathbf{x}_i, \mathbf{u}_i)} \ \ \forall i \in [1, N] \tag{16}$$

This allows rewriting the GAF energy in terms of a standard vectorial inner product:

$$E_{GAF} = \langle F, G \rangle = \sum_{i=1}^{N} F_i G_i = F^T G. \tag{17}$$

In summary, the notation changes as follows:

$$\begin{aligned} \mathbf{x} \in \Omega \subset \mathbb{R}^n &\to \mathbf{x} \in \mathbb{R}^{N \times n} \\ \mathbf{u}(\mathbf{x}) \in \mathbb{R}^p &\to \mathbf{u}(\mathbf{x}) \in \mathbb{R}^{N \times p} \\ f(\mathbf{x}, \mathbf{u}) \in \mathbb{R}_+ &\to F(\mathbf{x}, \mathbf{u}) \in \mathbb{R}_+^N \\ \sqrt{g(\mathbf{x}, \mathbf{u})} \in \mathbb{R}_+ &\to G(\mathbf{x}, \mathbf{u}) \in \mathbb{R}_+^N \\ E_{GAF}(\mathbf{u}) = \int_\Omega f \sqrt{g} \mathrm{d}\mathbf{x} &\to E'_{GAF}(\mathbf{u}) = F^T G \end{aligned}$$

E. Direct implementation for Cartesian images

In [10], a direct implementation of the GAF minimizing flow (14) is given by the following explicit (forward) Euler scheme, for each location $i \in [1, N]$ and deformation field component $p \in [1, p]$:

$$u_{i,q}^{k+1} = u_{i,q}^k + \Delta t \frac{\mathrm{d}u_{i,q}^k}{\mathrm{d}t}, \tag{18}$$

$$\frac{\mathrm{d}u_{i,q}^k}{\mathrm{d}t} = F_i H_{i,q} + \sum_{j=1}^{m=n+p} \left(D_j F_i \sum_{\mu,\nu=1}^{n} g_i^{\mu\nu} D_\mu X_i^j D_\nu u_{i,p}^k \right) - \frac{mn}{2\beta^2} D_{n+q} F_i \tag{19}$$

where $H \in \mathbb{R}^{N \times p}$ are the deformation field components of the embedded manifold's mean curvature vector, and $H_{i,q} = H_{i,q}(\mathbf{u}^k) \in \mathbb{R}$ denotes the component corresponding to $u_{i,q}$ at pixel i. The computation of H will be detailed below. $g_i^{\mu\nu} \in \mathbb{R}_+$ is the component μ, ν of the metric tensor at pixel i. $D_j F_i \in \mathbb{R}$ is the j-th component of the manifold gradient of F at pixel i. The first order differentials in space, e.g. $D_j F_i$ or $D_j u_{i,p}$, $j \in [1, n]$, are estimated using central differences, whereas the gradients w.r.t. the deformation field components, $D_{n+q} F_i$, $q \in [1, p]$, can be calculated analytically as a function of the estimated image gradients. More details are provided in [10].

F. Discretized Laplace-Beltrami operator

To estimate the Laplace-Beltrami operator on the deformation field, we use a second order central differences standard

scheme, e.g. [21]. To simplify the notation, we shorten the expression of the anisotropy tensor:

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix}_i = \sqrt{g_i} g_i^{\mu\nu}. \tag{20}$$

The scheme can now be written as a multiplication of a matrix $L \in \mathbb{R}^{N\times N}$ with the deformation field components $\mathbf{u} \in \mathbb{R}^{N\times p}$:

$$H = L\mathbf{u} \tag{21}$$

$$L_{i,j} = \frac{1}{\sqrt{g_i}} \cdot \begin{cases} -a(i) - c(i) & \\ \quad -\frac{1}{2}\left(a(i_{+,0}) + a(i_{-,0})\right) & \\ \quad -\frac{1}{2}\left(c(i_{0,+}) + c(i_{0,-})\right) & j = i \\ \frac{1}{2}\left(a(i) + a(j)\right) & j = i_{\pm_1,0} \\ \frac{1}{2}\left(c(i) + c(j)\right) & j = i_{0,\pm_2} \\ \frac{(\pm_1 1)\cdot(\pm_2 1)\left(b(i_{\pm_1,0}) + b(i_{0,\pm_2})\right)}{4} & j = i_{\pm_1,\pm_2} \end{cases} \tag{22}$$

where $i_{\pm,0} \mapsto (m \pm 1, n)$ denotes the index of the right/left-hand neighbor of pixel $i \mapsto (m, n)$, and $i_{0,\pm} \mapsto (m, n \pm 1)$, respectively. Further, $\pm_1$ versus $\pm_2$ are independent, whilst all $\pm_1$ are synchronous.

Based on the CFL condition [22], an upper bound on the time step that guarantees stability of such a scheme w.r.t. mean curvature vector flow, i.e. the first component of the complete GAF flow, is provided in [23]. In our context, this bound is given by

$$\Delta t \leq \frac{h^2}{8\beta \cdot \max_{m,n}\left\{\frac{a_{m\pm\frac{1}{2},n}}{\sqrt{g_{m,n}}}, \frac{c_{m,n\pm\frac{1}{2}}}{\sqrt{g_{m,n}}}\right\} \cdot \max\{F\}} \tag{23}$$

where $a_{m\pm\frac{1}{2},n} = \frac{1}{2}(a_{m,n} + a_{m\pm1,n})$ and $c_{m,n\pm\frac{1}{2}} = \frac{1}{2}(c_{m,n} + c_{m,n\pm1})$.

III. Lagrangian Multiplier

Lagrangian multiplier methods are a powerful and commonly used technique for constrained optimization. They are known to outperform ordinary penalty methods – where the constraint is only encouraged but not enforced [16], [24]. Indeed, a combination of both ordinary penalty and Lagrangian multiplier terms in the energy functional leads to the so-called Augmented Lagrangian (AL), that exhibits better convergence properties [14], [16].

To motivate the AL method briefly, let us consider a simple multidimensional, constrained optimization problem. Let $J, K \in \mathbb{R}$ be real scalar functions of the vector $\mathbf{x}$. J is the function to be optimized, and K represents the equality constraint to be satisfied:

$$\min_{\mathbf{x}} J(\mathbf{x}) \quad s.t. \quad K(\mathbf{x}) = 0. \tag{24}$$

A. Classical Lagrange multiplier

First, introducing a Lagrange multiplier $\lambda \in \mathbb{R}$ allows integrating the constraint within the Lagrangian function L:

$$L(\mathbf{x}, \lambda) = J(\mathbf{x}) + \lambda K(\mathbf{x}). \tag{25}$$

Now, it is easy to see that the solutions of (24) are located at the stationary points of the Lagrangian, i.e. where $\nabla_{\mathbf{x},\lambda} L = \mathbf{0}$. On the one hand, $\partial L/\partial\lambda = 0$ ensures the constraints are actually verified – i.e. the inner product vanishes –, on the other hand then the derivatives of L with respect to $\mathbf{x}$ need to be zero for J to be extremal.

B. Augmented Lagrangian

The augmented Lagrangian now consists in the introduction of both a Lagrange multiplier to enforce the constraints, and an ordinary quadratic penalty term that improves the convergence properties of some primal-dual minimization algorithms [14]:

$$AL(\mathbf{x}, \lambda, r) = J(\mathbf{x}) + \lambda K(\mathbf{x}) + \frac{r}{2} K(\mathbf{x})^2. \tag{26}$$

Given some initial guess for $\mathbf{x}^0$ and λ^0, the AL can now be minimized iteratively in three steps:

$$\begin{cases} \mathbf{x}^{k+1} = \operatorname{argmin}_{\mathbf{x}} AL(\mathbf{x}, \lambda^k, r^k), \\ \lambda^{k+1} = \lambda^k + \rho^k \cdot K(\mathbf{x}^{k+1}), \\ r^{k+1} \geq r^k, r^k \to \infty. \end{cases} \tag{27}$$

In [14], it was shown under the assumption K being a linear constraint then (27) converges for $r^k = r$ being a positive constant and $0 < \rho^k \leq 2r$. Thus we may choose $\rho^k = r$. Recent TV-regularized image processing problems, such as [25], [26], can be derived from such an AL scheme.

C. Dealing with multiple constraints

Let $K_i : \mathbb{R}^n \to \mathbb{R}, \ i = 1 \ldots m$ represent m multiple equality constraints. The augmented Lagrangian method can incorporate them all, by defining m corresponding Lagrangian multipliers λ_i:

$$AL(\mathbf{x}, \boldsymbol{\lambda}, r) = J(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i K_i(\mathbf{x}) + \frac{r}{2} \sum_{i=1}^{m} K_i(\mathbf{x})^2. \tag{28}$$

It is convenient to combine the m different constraint functions into a single, vector-valued constraint $\mathbf{K} : \mathbb{R}^n \to \mathbb{R}^m$, and to define a vectorial Lagrangian $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^m$. We may now rewrite the augmented Lagrangian incorporating multiple constraints as

$$AL(\mathbf{x}, \boldsymbol{\lambda}) = J(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{K}(\mathbf{x}) + \frac{r^k}{2} |\mathbf{K}(\mathbf{x})|^2, \tag{29}$$

where $|\mathbf{K}(\mathbf{x})|^2 = \mathbf{K}^T\mathbf{K}$

Assuming $\mathbf{K}$ being linear, the optimization scheme changes accordingly:

$$\begin{cases} \mathbf{x}^{k+1} = \operatorname{argmin}_{\mathbf{x}} AL(\mathbf{x}, \boldsymbol{\lambda}^k, r), \\ \boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \rho \cdot \mathbf{K}(\mathbf{x}^{k+1}). \end{cases} \tag{30}$$

IV. FastGAF Energy Minimization

The preliminary results presented in [10] were based on a direct implementation of the flow (7), using a simple forward Euler scheme. Here, we speed up the optimization task with a splitting approach, that minimizes the weighting function and the metric tensor term of the GAF energy separately, tightly coupled through an AL method.

Let us recall that $\mathbf{u} \in \mathbb{R}^{N\times p}$ is a matrix of size $N \times p$, where p is the co-dimension of the deformation field, i.e., the number of associated deformation field components.

A. Splitting

Firstly, we transform the unconstrained GAF energy minimization problem over one deformation field $\mathbf{u}$:

$$\min_{\mathbf{u}} \quad \left\{E_{GAF} = F(\mathbf{u})^T G(\mathbf{u})\right\} \tag{31}$$

into an equivalent, constrained minimization problem on two coupled deformation fields $\mathbf{u}$ and $\mathbf{v}$:

$$\min_{\mathbf{u},\mathbf{v}} \quad \left\{E_{GAF} = F(\mathbf{u})^T G(\mathbf{v})\right\} \quad s.t. \;\; \mathbf{u} = \mathbf{v}. \tag{32}$$

B. Augmented Lagrangian

We can obtain an unconstrained minimization problem and guarantee to satisfy the linear constraint $\mathbf{u} = \mathbf{v}$ using the following AL scheme [14], [16]:

$$\min_{\mathbf{u},\mathbf{v}} \left\{F(\mathbf{u})^T G(\mathbf{v}) + \langle \boldsymbol{\lambda}, \mathbf{u} - \mathbf{v}\rangle_{N,p} + \frac{r}{2}\|\mathbf{u} - \mathbf{v}\|_F^2\right\}, \tag{33}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{N\times p}$ is the Lagrangian multiplier matrix, in the same space as $\mathbf{u}$ and $\mathbf{v}$, $\langle \boldsymbol{\lambda}, \mathbf{u} - \mathbf{v}\rangle_{N,p} = \sum_{i=1}^{N}\sum_{d=1}^{p} \lambda_{i,d}(u_{i,d} - v_{i,d})$ is the scalar product between two matrices of the same dimension $N \times p$, and where $\|M\|_F^2 = \sum_{i,j} m_{i,j}^2$ is the square of the Matrix Frobenius norm. From now on, $r > 0$ is a positive constant; how to choose it will be discussed later.

Now, we apply algorithm (27) to solve the AL problem (33):

$$\begin{cases} (\mathbf{u}^{k+1}, \mathbf{v}^{k+1}) &= \operatorname{argmin}_{\mathbf{u},\mathbf{v}} \left\{F(\mathbf{u})^T G(\mathbf{v}) \right. \\ & \qquad \left. + \langle \boldsymbol{\lambda}^k, \mathbf{u} - \mathbf{v}\rangle_{N,p} + \frac{r}{2}\|\mathbf{u} - \mathbf{v}\|_F^2\right\}, \\ \lambda^{k+1} &= \lambda^k + \rho(\mathbf{u}^{k+1} - \mathbf{v}^{k+1}), \end{cases} \tag{34}$$

Further, this minimization w.r.t. $\mathbf{u}$ and $\mathbf{v}$ can be carried out separately, and we get the following split optimization scheme:

$$\begin{cases} \mathbf{u}^{k+1} &= \operatorname{argmin}_{\mathbf{u}} \left\{F(\mathbf{u})^T G(\mathbf{v}^k) \right. \\ & \qquad \left. + \langle \boldsymbol{\lambda}^k, \mathbf{u} - \mathbf{v}^k\rangle_{N,p} + \frac{r}{2}\|\mathbf{u} - \mathbf{v}^k\|_F^2\right\}, \\ \mathbf{v}^{k+1} &= \operatorname{argmin}_{\mathbf{v}} \left\{F(\mathbf{u}^{k+1})^T G(\mathbf{v}) \right. \\ & \qquad \left. + \langle \boldsymbol{\lambda}^k, \mathbf{u}^{k+1} - \mathbf{v}\rangle_{N,p} + \frac{r}{2}\|\mathbf{u}^{k+1} - \mathbf{v}\|_F^2\right\}, \\ \lambda^{k+1} &= \lambda^k + r(\mathbf{u}^{k+1} - \mathbf{v}^{k+1}), \end{cases} \tag{35}$$

In the next paragraphs, we will present how the two respective subminimization problems can be tackled efficiently.

C. Subminimization w.r.t. $\mathbf{u}$

The first subminimization problem in (35) deals with the optimization of the deformation field through the weighting function $F(\mathbf{u})$, i.e., the image distance function, while keeping the regularization term $G(\mathbf{v}^k)$ fix:

$$E_1(\mathbf{u}) = F(\mathbf{u})^T G(\mathbf{v}^k) + \langle \boldsymbol{\lambda}^k, \mathbf{u} - \mathbf{v}^k\rangle_{N,p} + \frac{r}{2}\|\mathbf{u} - \mathbf{v}^k\|_F^2. \tag{36}$$

To simplify things, we linearize F around the current estimate $F(\mathbf{u}^k)$. The Euler-Lagrange equation w.r.t. $\mathbf{u}$ is then obtained as:

$$0 = \operatorname{diag}(G(\mathbf{v}^k)) \cdot \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}^k) + \boldsymbol{\lambda}^k + r\mathbf{u} - r\mathbf{v}^k \tag{37}$$

$$\mathbf{u}^{k+1} = \mathbf{v}^k - \frac{1}{r}\left(\boldsymbol{\lambda}^k + \operatorname{diag}(G(\mathbf{v}^k)) \cdot \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}^k)\right). \tag{38}$$

Stability of this step is limited by the constant r: a small r results in a wider step away from the current estimate, and consequently the linear approximation of the image might not be good enough. We present a remedy in the next paragraph.

D. Balancing the computational complexity

The first order approximation requires small step sizes, i.e. big r. Further, as we will see below, the exact inverse solution of the second problem is computationally more challenging, but stable irrespective of the step size r. It would be interesting, both to balance the computational complexity between the two tasks, and to alleviate the step size restrictions in the first task. Here we propose a fixed-point scheme to adress these limitations. First, let us introduce a virtual time t and define a corresponding gradient descent equation:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{\partial E_1(\mathbf{u})}{\partial \mathbf{u}} \tag{39}$$

We discretize in time using a semi-implicit scheme:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\tau} = -\operatorname{diag}(G(\mathbf{v}^k)) \cdot \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}^n) - \boldsymbol{\lambda}^k - r\mathbf{u}^{n+1} + r\mathbf{v}^k \tag{40}$$

Using $\mathbf{u}^{n=0} = \mathbf{u}^k$ as initial condition, we iterate the scheme

$$\mathbf{u}^{n+1} = \frac{\mathbf{u}^n - \tau \operatorname{diag}(G(\mathbf{v}^k)) \cdot \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}^n) - \boldsymbol{\lambda}^k + r\mathbf{v}^k}{1 + \tau r} \tag{41}$$

until convergence towards a fixed point such that $\mathbf{u}^{k+1} = \mathbf{u}^{n\to\infty}$. We choose $\tau = \frac{1}{Lr}$, $L \in \mathbb{N}$, where we shorten the stepsize of the image disparity gradient by a factor L with respect to the direct solution (38). In practice, roughly $2L$ iterations are enough to achieve a satisfying convergence.

E. Subminimization w.r.t. $\mathbf{v}$

The second subminimization problem in (35) deals with the optimization of the deformation field through the regularization term $G(\mathbf{v})$, while, this time, keeping the weighting function term $F(\mathbf{u}^{k+1})$ fix:

$$E_2(\mathbf{v}) = F(\mathbf{u}^{k+1})^T G(\mathbf{v}) + \langle \boldsymbol{\lambda}^k, \mathbf{u}^{k+1} - \mathbf{v}\rangle_{N,p} + \frac{r}{2}\|\mathbf{u}^{k+1} - \mathbf{v}\|_F^2. \tag{42}$$

The derivative of the first term with respect to the deformation field $\mathbf{v}$ can be discretized in analogy to the Laplace-Beltrami operator:

$$\frac{\partial F(\mathbf{u}^{k+1})^T G(\mathbf{v})}{\partial \mathbf{v}} = -\operatorname{diag}(F(\mathbf{u}^{k+1})) \cdot W\mathbf{v} \tag{43}$$

where we introduce the Laplacian-like matrix $W \in \mathbb{R}^{N\times N}$ given by

$$W_{i,j} = 2\beta^2 \cdot \begin{cases} -a(i) - c(i) & \\ \quad -\frac{1}{2}\left(a(i_{+,0}) + a(i_{-,0})\right) & \\ \quad -\frac{1}{2}\left(c(i_{0,+}) + c(i_{0,-})\right) & j = i \\ \frac{1}{2}\left(a(i) + a(j)\right) & j = i_{\pm_1,0} \\ \frac{1}{2}\left(c(i) + c(j)\right) & j = i_{0,\pm_2} \\ \frac{(\pm_1 1)\cdot(\pm_2 1)\left(b(i_{\pm_1,0}) + b(i_{0,\pm_2})\right)}{4} & j = i_{\pm_1,\pm_2} \end{cases} \tag{44}$$

that differs from the Laplace-Beltrami discretization L only in its prefactor.

The complete Euler-Lagrange equation w.r.t. E_2 writes:

$$0 = -\operatorname{diag}(F(\mathbf{u}^{k+1}))\cdot W\mathbf{v} - \boldsymbol{\lambda}^k - r(\mathbf{u}^{k+1}-\mathbf{v}) \quad (45)$$

$$\mathbf{v}^{k+1} = \left(I - \frac{1}{r}\operatorname{diag}(F(\mathbf{u}^{k+1}))\cdot W\right)^{-1}\left(\mathbf{u}^{k+1} + \frac{1}{r}\boldsymbol{\lambda}^k\right) \quad (46)$$

Since we are only interested in the solution of this linear system, it is not necessary to explicitely compute the inverse of the right-hand-side matrix. Instead, more sophisticated solvers can exploit the sparsity of the system. We call this scheme the *exact inverse FastGAF*, in contrast to the approximate inverse FastGAF that will be introduced below. Note that this scheme highly resembles a semi-implicit Euler diffusion scheme, and therefore it can be considered unconditionally stable w.r.t. r [27].

F. A Jacobi scheme for approximate inversion

Instead of an exact solution, which is computationally expensive, we propose to use an iterative Jacobi scheme as an approximated solution, see e.g., [28]. Therefore, let us decompose the system matrix in a diagonal D and a remainder matrix R:

$$D + R = \left(I - \frac{1}{r}\operatorname{diag}(F(\mathbf{u}^{k+1}))\cdot W\right), \quad (47)$$

where D and R can be given explicitely:

$$\begin{aligned} D_{ii} &= 1 + \frac{2\beta^2 F_i}{r}\left(a(i)+c(i)\right) \\ &\quad + \frac{\beta^2 F_i}{r}\left(a(i_{+,0})+a(i_{-,0})\right) \\ &\quad + \frac{\beta^2 F_i}{r}\left(c(i_{0,+})+c(i_{0,-})\right) \end{aligned} \quad (48)$$

$$R_{ij} = \begin{cases} 0 & j = i \\ -\frac{\beta^2 F_i}{r}\left(a(i)+a(j)\right) & j = i_{\pm_1,0} \\ -\frac{\beta^2 F_i}{r}\left(c(i)+c(j)\right) & j = i_{0,\pm_2} \\ -\frac{(\pm_1 1)(\pm_2 1)\beta^2 F_i\left(b(i_{\pm_1,0})+b(i_{0,\pm_2})\right)}{2r} & j = i_{\pm_1,\pm_2} \end{cases} \quad (49)$$

Let us denote the right hand term $\mathbf{b} = \left(\mathbf{u}^{k+1} + \frac{1}{r}\lambda^k\right)$. Now, the Jacobi scheme approximates $\mathbf{v}^{k+1}$ as a fixed point solution, with $\mathbf{v}^{k+\frac{0}{J}} = \mathbf{u}^{k+1}$ as possible initialization:

$$(D+R)\mathbf{v} = \mathbf{b} \quad (50)$$

$$\mathbf{v}^{k+\frac{j}{J}} = \frac{\mathbf{b} - R\mathbf{v}^{k+\frac{j-1}{J}}}{D}, \quad j = 1,\ldots,J \quad (51)$$

where convergence can be guaranteed under two sufficient conditions:

- $D+R$ is strictly diagonally dominant, i.e. $\forall i : |D_i| > \sum_j |R_{ij}|$, or
- $D+R$ is definite positive, i.e. $\forall \mathbf{z} \neq \mathbf{0} : \mathbf{z}^T(D+R)\mathbf{z} > 0$.

In the present case, the first condition amounts to

$$\forall i: \quad r > \frac{\beta^2 F_i}{2}\sum_{\pm_1,\pm_2}\left(b(i_{\pm_1,0})+b(i_{0,\pm_2})\right), \quad (52)$$

i.e. convergence of the Jacobi scheme is not unconditionally guaranteed under the first condition. However, there is an easy upper bound on the step length $\frac{1}{r}$, but the system gets worse conditioned the closer one is to the true (rough) solution. Meanwhile, the second condition, i.e., definite positivity of the Laplacian, is provided and the scheme is expected to converge.

G. Complete approximate-inverse FastGAF algorithm

The complete FastGAF energy minimization algorithm now summarizes as follows:

$$\begin{cases} \mathbf{u}^{k+\frac{l+1}{2L}} = \dfrac{\mathbf{u}^{k+\frac{l}{2L}} - \frac{1}{Lr}\operatorname{diag}(G(\mathbf{v}^k))\cdot\frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}^{k+\frac{l}{2L}}) - \boldsymbol{\lambda}^k + r\mathbf{v}^k}{1+\tau r} \\ \qquad l = 0\ldots(2L-1) \\ \mathbf{v}^{k+\frac{j}{J}} = \begin{cases} \frac{\mathbf{u}^{k+1}+\frac{1}{r}\lambda^k - R\mathbf{u}^{k+1}}{D} & j = 1 \\ \frac{\mathbf{u}^{k+1}+\frac{1}{r}\lambda^k - R\mathbf{v}^{k+\frac{j-1}{J}}}{D} & j = 2\ldots J \end{cases} \\ \lambda^{k+1} = \lambda^k + r(\mathbf{u}^{k+1} - \mathbf{v}^{k+1}) \end{cases} \quad (53)$$

V. Experiments

We test the FastGAF scheme on two example applications: stereo vision disparity recovery, and 2D registration on a brain MRI slice. The results are compared to the direct implementation of the GAF minimizing flow, as exposed in [10], and the popular Demons algorithm, both in terms of registration error and time requirements. To illustrate the geometrical flexibility of the proposed FastGAF scheme, we show the results of a hemi-spherical toy-registration problem.

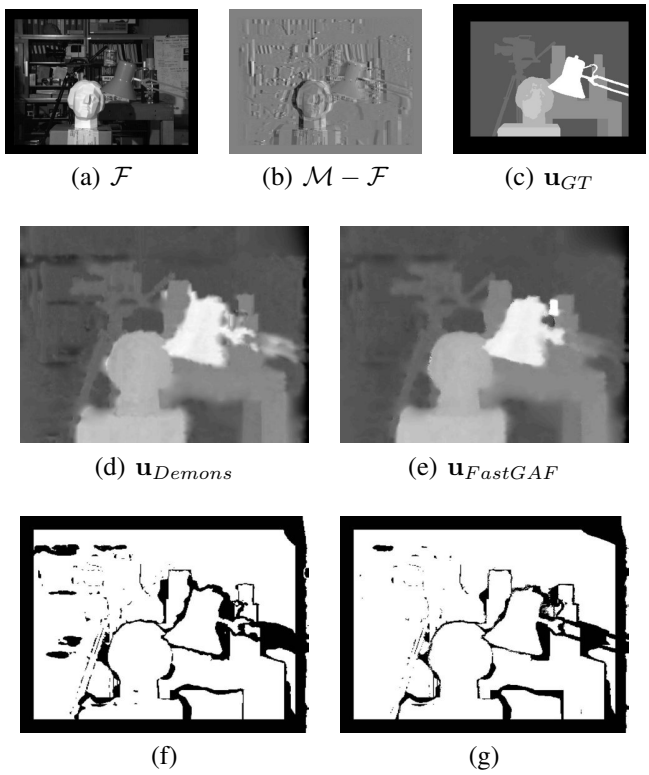
A. Multiresolution optimization and performance metrics

For all employed registration algorithms we employ the same coarse-to-fine multiresolution strategy. The images are low-pass filtered and subsampled at different resolutions and the minimization is executed in multiple stages. We select corresponding/matching parameters for all methods, in terms so of balancing α, metric β and step size r. We measure endpoint error EPE (in pixels) and percentage of wrong pixels $w1$ i.e., rate of pixels off by more than 1 pixel, as registration quality indicators. Each level stops after a predefinite numer of iterations K.

To measure the convergence speed of the different methods, we launch registration for a whole range of different numbers of iterations K, and plot the obtained respective quality measures against the wall clock time t required.

B. Stereo vision

An example of stereo vision depth recovery problem is shown in Fig. 1. The image pair `tsukuba` is a well known test image, taken from the middlebury benchmark set for stereo vision [20]. We are aware of the existence of a broad variety of high performance algorithms for the stereo vision depth recovery problem. Neither Demons, nor our proposed method are particularly well suited to compete with those state-of-the-art stereo vision methods. Here, we simply explore the stereo vision problem as an illustrative example due to its relatively simple embedding. The registration is set up according to the embedding described in (13) and using the absolute error weighting function (12).

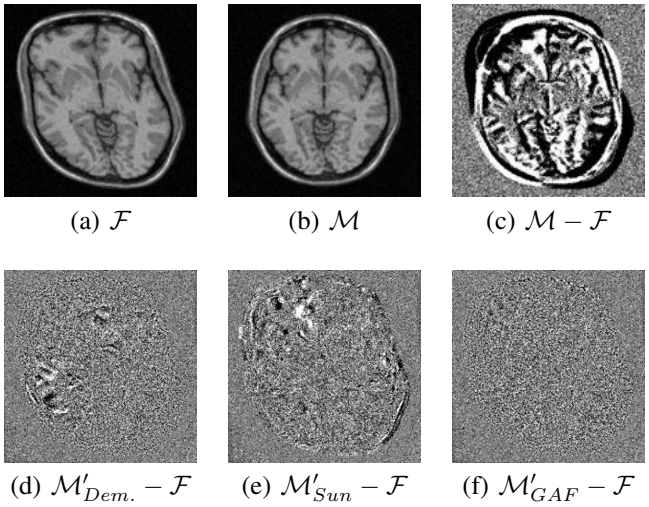


(a) $\mathcal{F}$ (b) $\mathcal{M} - \mathcal{F}$ (c) $\mathbf{u}_{GT}$

(d) $\mathbf{u}_{Demons}$ (e) $\mathbf{u}_{FastGAF}$

(f) (g)

Fig. 1. (a)–(b) The `tsukuba` test image for stereo vision and the image pair difference. (c) Ground truth disparity map. (d) State-of-the-art Demons registration. (e) Approximate-inverse FastGAF. (f)–(g) The location of wrong pixels (black) in the Demons and FastGAF result. Both methods primarily "fail" at occlusions.



(a) $\mathcal{F}$ (b) $\mathcal{M}$ (c) $\mathcal{M} - \mathcal{F}$

(d) $\mathcal{M}'_{Dem.} - \mathcal{F}$ (e) $\mathcal{M}'_{Sun} - \mathcal{F}$ (f) $\mathcal{M}'_{GAF} - \mathcal{F}$

Fig. 2. (a)–(b) Fix and moving image. (c) Intensity differences before registration (amplified). (d) Residue after registration with Demons, (e) Sun et al. [29], (f) FastGAF.

Registration takes place at 5 scales and we use the L_1 image distance metric. Parameters are chosen as $\alpha = 5$, $\beta^2 = 10$. For the exact-inverse FastGAF scheme, we use $L = 100$ to balance the computational load between the two subminimization steps, whereas with the approximate-inverse scheme with $J = 4$ Jacobi iterations only, $L = 4$ seems more appropriate.

As shown in Fig. 1, the proposed FastGAF scheme outperforms the state-of-the-art Demons method.

C. 2D registration

The second case deals with 2D registration of a highly misaligned monomodal medical image pair. An axial slice through a T1 MRI volume is heavily deformed by a given 2D deformation field. The initial average endpoint error is 7.3 pixels. The images have a resolution of 317×317 pixels and are both affected by 5% additive Gaussian noise. Registration is set up with the squared error weighting function (11). The image pair and initial error are illustrated in Fig. 2, along with the respective results of the Demons and FastGAF registration. We also compare against a very competitive optical-flow method by Sun et al., that uses a fast non-local regularization approach [29].

For the brain registration case we use 5 scales and employ the L_2 distance measure. Further parameters are $\alpha = 50$, $\beta^2 = 3$. With the exact inversion scheme, we use $L = 100$, whereas with approximate inversion we chose $L = 10$ for $J = 4$ Jacobi iterations.

Demons converges to a wrong-pixel rate of $w1 = 13.2\%$ within $t = 664$, corresponding to a final $EPE = 0.515$. The optical flow method converges considerably faster within $t = 26$ seconds, yielding $w1 = 13.3\%$ and $EPE = 0.547$. The proposed FastGAF scheme requires $t = 84$ to obtain $w1 = 6.8\%$, $EPE = 0.327$.

D. Hemi-spherical registration

One of the main strength of the Geodesic Active Fields method for image registration, is its intrinsic ability to deal with non-Euclidean images. Here we show an example of a spherical image, parametrized through stereographic projection. While we use the well-known topography of the Earth as a toy example, realistic applications can be found in omnidirectional vision e.g., [30]. Indeed, in [31], [32] it was shown, that a sensor image of a catadioptric camera (using spherical, paraboloid or hyperboloid mirror) is equivalent to a stereographic projection of the spherical plenoptic function.

The metric of the stereographic projection is conformal to the regular Euclidean metric [33]. The choice of the metric h_{ij} of the embedding space is immediate:

$$h_{ij} = \frac{4}{1 + x^2 + y^2} \operatorname{diag}(1, 1, \beta^2, \beta^2) \tag{54}$$

Three typical stereographic projections, polar, equatorial and oblique, are sketched in Fig. 3(a)–(c). Here we chose an oblique projection of the Earth's topographical map for the sake of generality. The map is artifically deformed and sucessfully restored through registration, see Fig. 3(d)–(f).

VI. Discussion

Qualitative results of the registration process using the proposed approximate-inverse FastGAF scheme and the state-of-the-art Demons method were shown in fig. 1 and fig. 2, respectively. Here, a more quantitative view of the respective

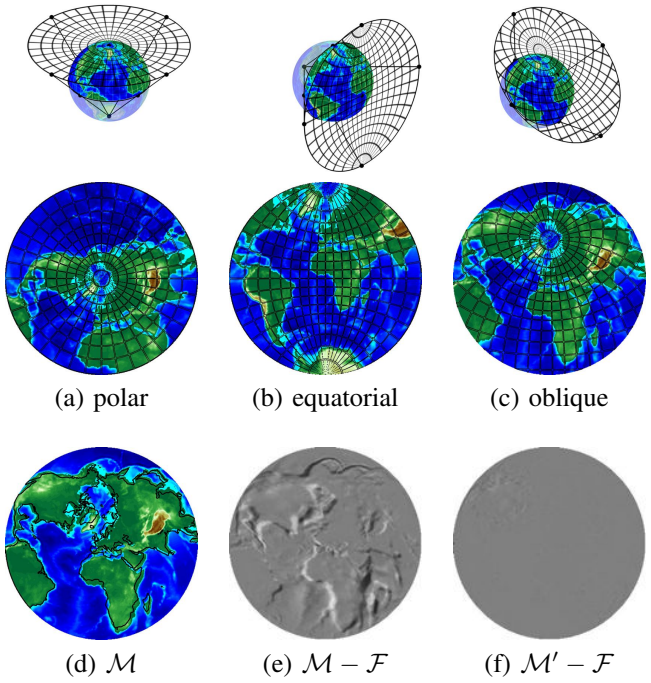


Fig. 3. (a)–(c) Different stereographic projections of the sphere. (d) Chosen oblique projection and artificial deformation (original coastlines in black). (e)–(f) Intensity differences before and after registration.

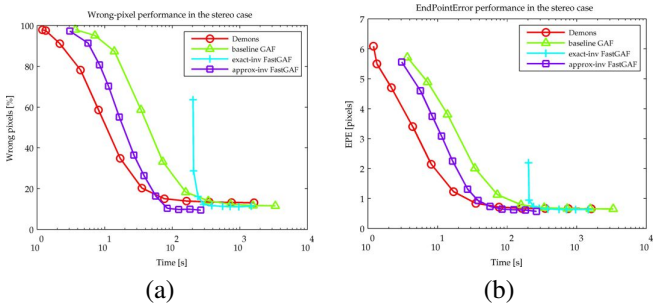


Fig. 4. Comparison of different registration schemes for stereo depth recovery. Demons (red, circles), baseline GAF (green, triangles), exact-inverse FastGAF (cyan, plus), approximate-inverse FastGAF (blue, squares). (a) Percentage of wrong pixels (i.e. more than 1 pixel off). (b) rMSE on the recovered disparity field.

algorithm performances is given in fig. 4 and fig. 5. In the next few paragraphs, we first discuss the performance differences between the four methods considered and highlight the advantages of the proposed scheme. Then, we have a closer look at the different roles of the parameters involved in the FastGAF scheme. In particular, we highlight the importance of an augmented Lagrangian scheme for constraint internalization, compared to pure Lagrangian or penalty deformation field coupling.

A. Algorithms and performance comparison

The four methods compared, i.e., the direct, baseline GAF implementation, the state-of-the-art Demons method, and the proposed accurate- and approximate-inverse FastGAF minimization, share a lot of common features, but have different parameters and stability requirements.

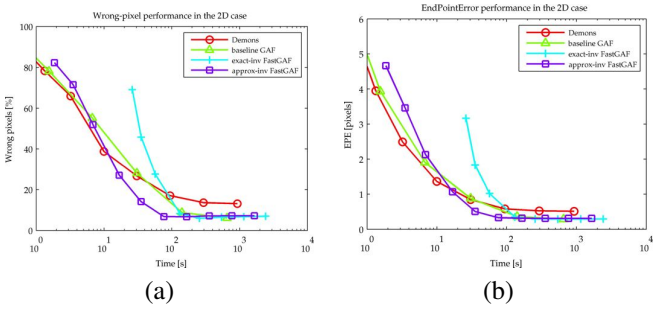


Fig. 5. Comparison of different registration schemes for 2D image registration. Demons (red, circles), baseline GAF (green, triangles), exact-inverse FastGAF (cyan, plus), approximate-inverse FastGAF (blue, squares). (a) Percentage of wrong pixels (i.e. more than 1 pixel off). (b) rMSE on the recovered disparity field.

The main commonalities and differences can be resumed as follows:

- With Demons, the step size, i.e. the number of iterations required until convergence, is essentially determined by the desired smoothness of the deformation field. While the smoothing kernel g is fixed, preferably Gaussian, the smoothness is entirely determined by the relative stepsize of the data-term, governed by both r and $1/\sigma_i^2$. In this respect, σ_i^2 and σ_T^2 are actually redundant.
- The step size of the direct GAF implementation is heavily limited by the stability criterion of the mean curvature flow component. Note that the criterion is dependant both on the range of the involved metric, and the gradients of the deformation field. The expected, i.e. non-worst-case stability criterion is actually observed well less restrictive, and a β^2 dependancy, rather than β^3 can be realistically assumed. Also, the effect of the central differences approximation of advective middle term in the minimizing flow is highly diffusive. Additionally, in contrast to the two other schemes, the curvature flow does never act on its own, but always simultaneously with the advective and data term, which generally acts in favor of stability.
- The proposed accurate-inverse FastGAF scheme offers unconditional stability with respect to the curvature flow. The maximum progression rate is only limited by the stability of the data term, which is greatly enhanced thanks to the L-fold subdivision of the data step. The number of required iterations is therefore reduced importantly, at the expense of higher cost per iteration due to the sparse matrix inversion.
- The latter cost can be drastically reduced by employing an approximate inversion scheme, as proposed with the approximate-inverse FastGAF scheme.

Overall, the approximate-inverse FastGAF scheme can provide the various benefits of the geometric GAF framework for image registration in very attractive computational times, comparing favorably to state-of-the-art methods.

B. The roles of the parameters r, L and J

The approximate-inverse FastGAF scheme introduces three new parameters to the problem. Here we want to illustrate

and discuss their respective role in the optimization framework from a performance-based point of view.

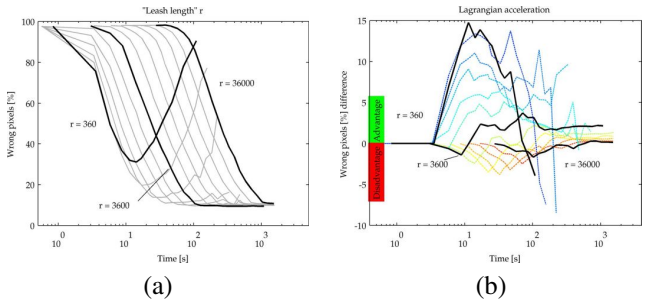
The first novel parameter introduced with the augmented Lagrangian scheme is the penalty weight r. The penalty term is supposed to stabilize the minimization problem. Indeed, it introduces a quadratic energy on the distance between the separate split deformation fields and acts as a "leash" between the two, its elasticity being governed by r. The bigger this parameter, the closer u and v are tied together and the less they can diverge at each iteration. This reduces the lag between u and v and is the only way to restrict the step size at each iteration. The step size limitation is critical with respect to the data term, as due to the complexity of the images under consideration, the linear approximation of the weighting function only holds within a close vicinity of the current configuration. The impact of r on the registration perfomance is illustrated in fig. 6. It becomes clear, that below a certain level the optimization is unstable with respect to the data term, whereas above a certain threshold only the speed of the algorithm is affected. In particular, it becomes obvious that the penalty term itself is vital for stability, and a pure Lagrangian approach implies severe convergence issues. In contrast, the Lagrangian multiplier is not an absolute requirement for the split GAF iterations to converge, as illustrated in fig. 6b). However, as can be seen from the charts, it accelerates the convergence of the algorithm especially for low penalty weight and thus confirms to be of true interest, considering its little computational extra effort.

In order to overcome the small-steps requirement of the data-term, we propose to carry out several data-step optimizations within a fixed-point scheme before doing a more important smoothing step. The number of data-term optimizations per regularity-term optimization is governed by the parameter L. At r fixed, the image distance term is optimized more precisely, while by replacing $r \leftarrow r/L$, the leash length can be extended at the same image distance precision, thus reducing the number of smoothing steps required. This speed gain is illustrated in fig. 7. In particular with the exact-inverse FastGAF scheme, this parameter can be seen as a way of balancing the computational load between both optimization subtasks, to gain in efficiency.

In the approximate-inverse FastGAF scheme, finally, we replace the costly full inversion of the diffusion matrix by an approximating Jacobi scheme. There, the number of Jacobi iterations is given by J, controlling the desired precision of the approximate inverse. While the exact inverse has full support (full matrix), now the support of the smoothing stencil is of size $2J + 1$ in each dimension and the speedup is obviously considerable. The perfomance for different J can be seen in fig. 8. Close to the critical step-length r, the algorithm does not properly converge for small, odd, J. Otherwise, the speedup comes at little loss of precision, which is reasonable since the smoothing is essentially operating in a closed loop.

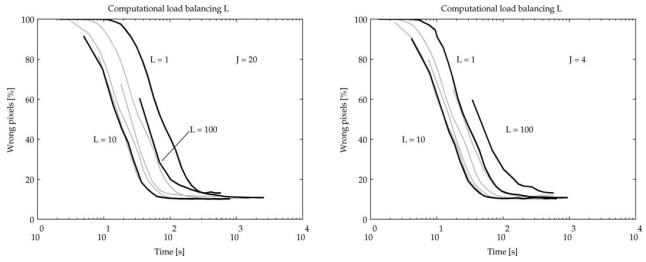
VII. Conclusions

In this paper, we have presented a splitting scheme for GAF, based on the method of augmented Lagrangians. The



(a) (b)

Fig. 6. Influence of the penalty term and the Lagrangian multiplier on the registration performance. (a) Smaller r increases the learning rate, i.e. the step length, at the cost of stability due to a too big data step (here at $r = 500$). Above a certain level, once stability achieved, increasing r only scales the computation time without gaining in precision. (b) Compared to penalty-term only, the Lagrangian multiplier reduces the wrong-pixels rate early in the optimization, in particular for low r. Registration converges to the same results, but more quickly.



(a) (b)

Fig. 7. Compuational load balancing L. Subdividing the data-step allows reducing r by L while roughly keeping the same data-stability. This can provide speedups, as in effect a smoothing step (and Lagrangian update) is only performed after L data steps. Relatively small load balancing factors, here about $L = 10$ yields best performance. Beyond, e.g. at $L = 100$, the speed gain becomes negligible while the regularization step is performed too seldom and registration quality is slightly decreased. (a) The speedup is important for computationally heavy smoothing task $J = 20$ (or exact inverse). (b) For lightweight smoothing, $J = 2$, the speed gains are less pronounced.

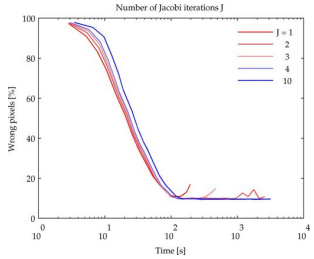


Fig. 8. Number of iterations J of the Jacobi approximate inversion scheme. While too coarse approximations for $J = 1$ seem to be unsufficient in the present case, $J > 1$ all yield similar, good registration quality, while scaling the computational time accordingly.

optimization takes place in two alternating steps. In the first problem, the data-term is optimized using a fixed-point scheme derived from a semi-implicit gradient-descent discretization. The second task optimizes with respect to the smoothing-term, initially by exactly solving a sparse linear system, a method we call *exact-inverse FastGAF*. Then, we substitute with an approximate inversion within the semi-implicit smoothing step, using a few fast Jacobi iterations only, to obtain the *approximate-inverse FastGAF* scheme. This scheme results in a considerable speedup of the registration process with respect to the baseline GAF as introduced in [10], and also compares favorably to the Demons state-of-the-art registration method. While the Jacobi scheme already provides a nice speedup, more recent and efficient methods e.g., additive operator splitting (AOS) may further improve performance [27], [34]–[36].

From a more fundamental perspective, geodesic active fields can be considered a generalization of the Demons method in several respects. First, the GAF framework is designed to work on Riemannian manifolds and is thus not restricted to Euclidean images. Although Demons can be generalized to non-flat images as well, e.g. [37], on non-Cartesian grids the speed advantages of Gaussian convolution are lost. While Demons regularization is explicitely Gaussian – it penalizes the L_2-norm of the deformation field gradient – Beltrami regularization offers a tunable interpolation between Gaussian L_2 and more anisotropic, TV-norm like L_1 regularization. Also, the GAF framework offers the advantage of being parametrization invariant. Finally, the GAF registration framework comes with no preferred image discrepancy measure, whereas Demons has a strong preference for the L_2-norm on the image differences (SSD). With the approximate-inverse FastGAF scheme presented in this paper, we are able to achieve those relative advantages in very competitive computation times.

REFERENCES

[1] L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surveys*, vol. 24, no. 4, pp. 325–376, Dec. 1992.

[2] J. B. A. Maintz and M. A. Viergever, "A survey of medical image registration," *Med. Image Anal.*, vol. 2, no. 1, pp. 1–36, Mar. 1998.

[3] M. A. Audette, F. P. Ferrie, and T. M. Peters, "An algorithmic overview of surface registration techniques for medical imaging," *Med. Image Anal.*, vol. 4, no. 3, pp. 201–17, 2000.

[4] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image Vision Comput.*, vol. 21, no. 11, pp. 977–1000, Oct. 2003.

[5] G. Hermosillo, C. Chefd'hotel, and O. Faugeras, "Variational methods for multimodal image matching," *Int. J. Comput. Vis.*, vol. 50, no. 3, pp. 329–343, 2002.

[6] J.-P. Thirion, "Image matching as a diffusion process: an analogy with Maxwell's demons," *Med. Image Anal.*, vol. 2, no. 3, pp. 243–260, 1998.

[7] A. N. Tichonov, "Solution of incorrectly formulated problems and the regularization method," *Soviet Mathematics*, vol. 4, pp. 1035–1038, 1963.

[8] V. A. Morozov, "Linear and nonlinear ill-posed problems," *Journal of Mathematical Sciences*, vol. 4, no. 6, pp. 706–736, 1975.

[9] T. Vercauteren, X. Pennex, E. Malis, A. Perchant, and N. Ayache, "Insight into efficient image registration techniques and the demons algorithm," in *Information Processing in Medical Imaging*, 2007, pp. 495–506.

[10] D. Zosso, X. Bresson, and J.-P. Thiran, "Geodesic active fields - a geometric framework for image registration," *IEEE Trans. Image Process.*, 2010, in press.

[11] N. Sochen, R. Kimmel, and R. Malladi, "A general framework for low level vision," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 310–318, Mar. 1998.

[12] A. M. Polyakov, "Quantum geometry of bosonic strings," *Phys. Lett. B*, vol. 103, no. 3, pp. 207–210, 1981.

[13] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comput. Vis.*, vol. 22, no. 1, pp. 61–79, 1997.

[14] R. Glowinski and P. Le Tallec, *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*. Philadelphia: Society for Industrial and Applied Mathematics (SIAM), 1989.

[15] P. Lions and B. Mercier, "Splitting algorithms for the sum of two nonlinear operators," *SIAM J. Numer. Anal.*, vol. 16, no. 6, pp. 964–979, Dec. 1979.

[16] J. Nocedal and S. Wright, *Numerical optimization*, 2nd ed. Springer, Berlin, 2006.

[17] X. Bresson, P. Vandergheynst, and J.-P. Thiran, "Multiscale active contours," *Int. J. Comput. Vis.*, vol. 70, no. 3, pp. 197–211, Dec. 2006.

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

[19] D. Zosso, X. Bresson, and J.-P. Thiran, "Geodesic active fields - a geometric framework for image registration," Ecole Polytechnique Fédérale de Lausanne (EPFL), Tech. Rep. LTS-REPORT-2010-001, 2010.

[20] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1, pp. 7–42, 2002.

[21] R. Ben-Ari and N. Sochen, "A geometric framework and a new criterion in optical flow modeling," *J. Math. Imaging Vis.*, vol. 33, no. 2, pp. 178–194, Feb. 2009.

[22] R. Courant, K. Friedrichs, and H. Lewy, "Über die partiellen Differenzengleichungen der mathematischen Physik," *Mathematische Annalen*, vol. 100, no. 1, pp. 32–74, 1928.

[23] L. Dascal and N. A. Sochen, "A maximum principle for Beltrami color flow," *SIAM Journal on Applied Mathematics*, vol. 65, no. 5, pp. 1615–1632, 2005.

[24] D. P. Bertsekas, "Multiplier methods: A survey," *Automatica*, vol. 12, no. 2, pp. 133–145, 1976.

[25] X.-C. Tai and C. Wu, "Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model," in *Scale Space and Variational Methods in Computer Vision*, 2009, pp. 502–513.

[26] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[27] J. Weickert, B. M. ter Haar Romeny, and M. A. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 398–410, Mar. 1998.

[28] D. B. Duncan and M. A. M. Lynch, "Jacobi iteration in implicit difference schemes for the wave equation," *SIAM Journal on Numerical Analysis*, vol. 28, no. 6, pp. 1661–1679, 1991.

[29] D. Sun, S. Roth, and M. J. Black, "Secrets of optical flow estimation and their principles," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, Jun. 2010, pp. 2432 –2439.

[30] I. Tosic, I. Bogdanova, P. Frossard, and P. Vandergheynst, "Multiresolution motion estimation for omnidirectional images," in *Proceedings of 13th EUSIPCO*, 2005.

[31] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems and practical implications," in *Computer Vision ECCV 2000*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2000, vol. 1843, pp. 445–461.

[32] ——, "Catadioptric projective geometry," *International Journal of Computer Vision*, vol. 45, pp. 223–243, 2001.

[33] I. Bogdanova, X. Bresson, J.-P. Thiran, and P. Vandergheynst, "Scale Space Analysis and Active Contours for Omnidirectional Images," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1888–1901, Jul. 2007.

[34] J. Weickert, "A review of nonlinear diffusion filtering," in *Proc. Scale-Space Theory Computer Vision*, ser. Lecture Notes in Computer Science, vol. 1252. Berlin, Germany: Springer-Verlag, 1997, pp. 3–28.

[35] R. Malladi and I. Ravve, "Fast difference schemes for edge enhancing Beltrami flow," in *ECCV 2002*, ser. Lecture Notes in Computer Science, vol. 2350, 2002, pp. 343–357.

[36] L. Dascal, G. Rosman, X.-C. Tai, and R. Kimmel, "On semi-implicit splitting schemes for the Beltrami color flow," in *Scale Space and Variational Methods in Computer Vision*, ser. Lecture Notes in Computer Science, X.-C. Tai, K. Mrken, M. Lysaker, and K.-A. Lie, Eds., vol. 5567. Springer Berlin / Heidelberg, 2009, pp. 259–270.

[37] B. Yeo, M. Sabuncu, T. Vercauteren, N. Ayache, B. Fischl, and P. Golland, "Spherical demons: Fast diffeomorphic landmark-free surface registration," *IEEE Trans. Med. Imag.*, vol. 29, no. 3, pp. 650 –668, Mar. 2010.