SIAM J. SCI. COMPUT.
Vol. 27, No. 5, pp. 1594–1607

A MULTILEVEL METHOD FOR IMAGE REGISTRATION*

ELDAD HABER† AND JAN MODERSITZKI†

Abstract. In this paper we introduce a new framework for image registration. Our formulation is based on consistent discretization of the optimization problem coupled with a multigrid solution of the linear system which evolves in a Gauss–Newton iteration. We show that our discretization is h-elliptic independent of parameter choice, and therefore a simple multigrid implementation can be used. To overcome potential large nonlinearities and to further speed up computation, we use a multilevel continuation technique. We demonstrate the efficiency of our method on a realistic highly nonlinear registration problem.

Key words. image registration, multigrid, multilevel, optimization

AMS subject classifications. 92C55, 65M55, 15A23, 65K10

DOI. 10.1137/040608106

1. Introduction and problem setup.

Image registration is one of today's challenging image processing problems. Given a so-called reference R and a so-called template image T, the basic idea is to find a *reasonable* transformation such that a transformed version of the template image becomes *similar* to the reference image. Image registration has to be applied whenever images resulting from different times, devices, and/or perspectives need to be compared or integrated; see, e.g., [6, 19] and references therein.

The computation of nonrigid image registration has two main building blocks. The first one is a distance measure $\mathcal{D}$ quantifying distance or similarity of images, and the second one is a regularizer $\mathcal{S}$ which rules out nonreasonable solutions. Note that image registration is an ill-posed problem; see, e.g., [19]. Therefore, regularization is inevitable. A common treatment of the registration problem is based on the following variational approach: Find a transformation $\boldsymbol{u}$ minimizing the joint energy

$$\mathcal{J}^\alpha[\boldsymbol{u}] := \mathcal{D}(R, T(\boldsymbol{u})) + \alpha\mathcal{S}(\boldsymbol{u}). \tag{1}$$

Here, $\alpha > 0$ is a regularization parameter that compromises similarity and regularity.

The formulation (1) is common to many ill-posed inverse problems where the data fitting term $\mathcal{D}$ is balanced with a regularization term using the regularization parameter α. Many authors have dealt with the optimal choice of α, using a variety of approaches; see, for example, [14, 24]. If the difference between the images is associated with random noise, then standard methods that are based on the statistics of the noise, such as generalized cross validation (GCV) [9] or χ^2 [21], can be used. However, in image registration the difference between two images (at least in most medical applications) is not associated with random noise. In fact, the difference may be highly structured, and only a trained professional (e.g., a radiologist) could determine the exact amount of regularization needed. Therefore, we prefer to provide a sequence of registered image pairs for various α's and leave the final choice of an optimal α to a trained professional.

*Received by the editors May 12, 2004; accepted for publication (in revised form) January 3, 2005; published electronically February 3, 2006. This research was supported by NSF grant CCF-0427094.
http://www.siam.org/journals/sisc/27-5/60810.html
†Department of Mathematics and Computer Science, Emory University, Atlanta GA 30322 (haber@mathcs.emory.edu, modersit@mathcs.emory.edu).

In general, the optimization problem (1) cannot be solved analytically. Thus, numerical schemes and appropriate discretizations are required. Typically, a discretization leads to a finite dimensional nonlinear problem, where the number of unknowns can be large. For example, the discrete registration problem for two $128 \times 128 \times 64$ MRI scans results in a nonlinear system of equations in about $3 \cdot 10^6$ unknowns. Thus, highly efficient algorithms are required in order to perform the registration in a reasonable amount of time. This applies, in particular, if one is interested not only in a solution for a fixed α but in a sequence of solutions for a variety of different α's. Furthermore, since we want our codes to run in a clinical setting, we require only modest computational hardware.

There are two main approaches for the discretization of the registration problem (1). In the first, the *optimize-discretize* approach, one forms the objective function and then differentiates to obtain the continuous Euler–Lagrange equations, which are finally discretized and solved numerically; see, e.g., [7, 15, 19]. The second approach is the *discretize-optimize* approach. Here, one directly discretizes the problem and then solves a finite (but typically high) dimensional optimization problem; see, e.g., [13]. The advantage of the latter approach is that standard optimization methods can be used. We prefer the discretize-optimize approach; however, in order to take advantage of efficient optimization techniques, all parts of the discrete problem need to be continuously differentiable. We discuss the implications and advantages of this necessity in the paper.

To gain additional computational efficiency and obtain a scalable algorithm, one can either use a full approximation multigrid scheme (FAS) or a multilevel Newton-type scheme with an efficient solver for the linearized systems of equations. In this paper we propose a multilevel inexact Gauss–Newton scheme combined with a multigrid solver for a computation of a numerical solution of (1). Although no proof exists, the latter approach has an efficiency similar to that of FAS [23, p. 154], but it is much more modular. Particularly, the linear solver is decoupled from the optimization strategy. Therefore, the code can be modified easily to deal with any differentiable distance measure and/or regularizer. However, for the sake of simplicity, we focus on the so-called *sum of squared differences* (SSD) as a distance measure $\mathcal{D}$ and the elastic potential as a regularizer $\mathcal{S}$:

$$\mathcal{D}[\boldsymbol{u}] = \frac{1}{2}\int_\Omega (T(\boldsymbol{x}+\boldsymbol{u}(\boldsymbol{x})) - R(\boldsymbol{x}))^2 \, d\boldsymbol{x}, \tag{2a}$$

$$\begin{aligned}\mathcal{S}[\boldsymbol{u}] &= \int_\Omega \langle \mathcal{B}\boldsymbol{u}, \mathcal{B}\boldsymbol{u}\rangle \; d\boldsymbol{x} \\ &:= \frac{1}{2}\left\| \begin{pmatrix} \sqrt{\lambda+\mu}\,\nabla\cdot & & \\ \sqrt{\mu}\,\nabla & & \\ & \ddots & \\ & & \sqrt{\mu}\,\nabla \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{u}_d \end{pmatrix} \right\|^2_{L_2(\Omega)} \\ &= \int_\Omega \frac{\lambda+\mu}{2} \| \nabla \cdot \boldsymbol{u}\|^2 + \frac{\mu}{2}\sum_{i=1}^{d} \| \nabla \boldsymbol{u}_i\|^2 \, d\boldsymbol{x},\end{aligned} \tag{2b}$$

where $\Omega = (0,1)^d$ is the registration domain and λ and μ are the Lamé constants.

Our discretization is based on the weak form of the problem (1) which naturally leads to staggered grid discretization. To the best of our knowledge, staggered grid discretization was introduced into imaging in [13]. In this paper, we prove the h-ellipticity of the discretization for the elastic potential, independent of the problem parameters. As has already been shown in [13], this discretization is also stable for pointwise volume preserving constraints.

Our work relates to the work of Henn and Witsch [15], Clarenz, Droske, and Rumpf [7], and Kalmoun and Rüde [18]. In [15], an FAS based on the discretized optimality conditions for the continuous problem is presented. In [7], a diffusive rather than an elastic regularizer is used. A d-linear interpolation scheme is used to compute the deformed template in [7, 15, 18]. Therefore, the objective function is not continuously differentiable, and this may lead to failure of standard optimization techniques. In [18] a multigrid method is developed for the related optical flow problem but with the diffusive regularizer.

This paper is organized as follows. In section 2, we discuss the discretization of the registration problem (1). Our optimization approach, which is based on a Gauss–Newton-type scheme (cf., e.g., [20]), is discussed in section 3. In section 4, we propose a multigrid method for a solution of the linearized problems. Using a local mode analysis we show that our discretization is particularly amenable to multigrid methods and suggest appropriate multigrid treatment. In section 5, we combine our nonlinear iteration with a multilevel continuation method. Finally, in section 6, we present numerical examples that demonstrate the effectiveness of our algorithm.

2. Discretization. Choosing a stable discretization method for a system of partial differential equations with mixed derivatives is a delicate matter. Here, we use staggered grids (cf. Figure 1), which are very common for stable discretizations of fluid flow (see, e.g., [8]) and electromagnetics (see, e.g., [11, 27]), where operators such as the gradient, curl, and divergence are discretized. It is also well known that staggered grids are tightly connected to mixed finite element methods which are commonly used for elasticity [3].

In this section we briefly summarize the discretization we use. Further discussion and details are given in [13].

2.1. Discretizing $\boldsymbol{u}$. We assume that our discrete images have $m_1 \times \cdots \times m_d$ pixels, where $d = 2, 3$ is the image dimensionality. For ease of presentation, we also assume that each pixel is square or cubic, where each side has length h. In our description we allow for half-step indices. As usual in image processing, we identify pixels/voxels with cell centered grid points $x_{j,k,\ell}$, which are therefore labeled with full integer indices. Given a pixel/voxel $x_{j,k,\ell}$, their faces are numbered with a half index, $x_{j\pm\frac{1}{2},k,\ell}$, $x_{j,k\pm\frac{1}{2},\ell}$, and $x_{j,k,\ell\pm\frac{1}{2}}$, and we discretize the ith component u^i of $\boldsymbol{u}$ on the ith face for every pixel/voxel. With some abuse of notation, we denote the discrete analogue of the continuous vector field by $\boldsymbol{u} = (u^1, \dots, u^d)^\top$, where u^i denotes the grid function which is approximated on the face staggered grid.

If needed, the derivatives $\partial_j u^k$ are approximated by the short (central) differences,

$$\partial_j u^k \approx \partial_j^h u^k := \frac{1}{h}\Big(u^k_{\dots,i_j+\frac{1}{2},\dots} - u^k_{\dots,i_j-\frac{1}{2},\dots}\Big). \tag{3}$$

Note that no boundary conditions are needed to approximate derivatives in the normal directions ($\partial_j u^j$). For the tangential directions ($\partial_j u^k$, $j \neq k$) we imposed Neumann boundary conditions.

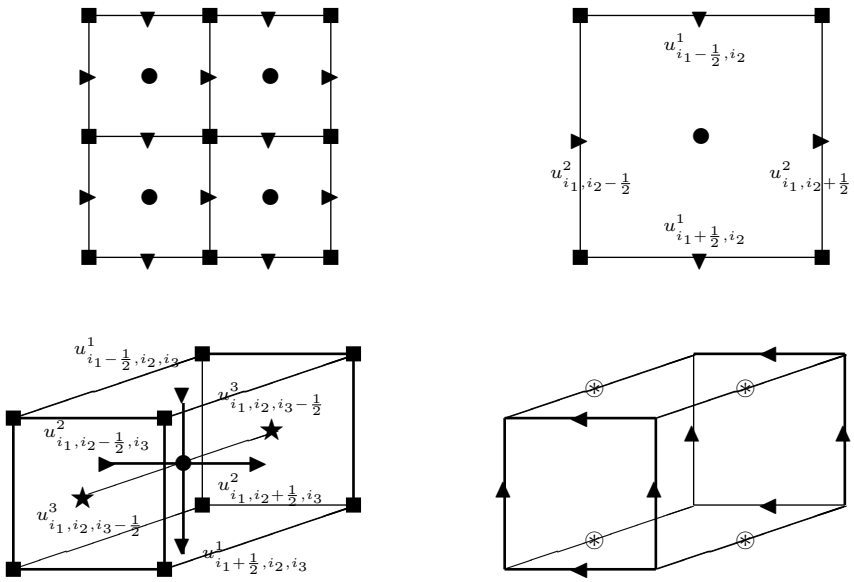

FIG. 1. *Staggered grids for dimensions $d = 2, 3$, nodal ■, cell centered •, face staggered grids (▼ in x_1-, ► in x_2-, and ★ in x_3-direction), and edge staggered grids (▲ in x_1-, ◄ in x_2-, and ⊛ in x_3-direction). Top left: $d = 2$, four pixels. Top right: $d = 2$ pixel (i_1, i_2) with grids. Bottom left: $d = 3$, voxel (i_1, i_2, i_3) with face staggered grids and positions of u^1, u^2, u^3. Bottom right: $d = 3$, edge staggered grids.*

2.2. Discretizing $\mathcal{S}$. Since many regularizers are phrased in terms of the more complex differential operators gradient ∇ and divergence $\nabla\cdot$, we introduce the notation ∇_j^h and $\nabla^h\cdot$ for the discrete analogues,

$$\nabla_j^h\, u^j = (\partial_1^h u^j, \ldots, \partial_d^h u^j)^\top \quad \text{and} \quad \nabla^h \cdot \boldsymbol{u} = \sum \partial_j^h u^j. \tag{4}$$

Using these discrete analogues the elastic potential $\mathcal{S}$ (2b) is discretized as

$$\mathcal{S}^h(\boldsymbol{u}) = \|\boldsymbol{B}\boldsymbol{u}\|_2^2 := \frac{\lambda+\mu}{2}\| \nabla^h \cdot \boldsymbol{u}\|^2 + \frac{\mu}{2}\sum_{i=1}^{d} \| \nabla_j^h \boldsymbol{u}_i\|^2. \tag{5}$$

In the course of the registration process we require derivatives. Upon differentiation of the regularizer we obtain the Navier–Lamé operator

$$\mathcal{S}_{\boldsymbol{u}}^h(\boldsymbol{u}) = (\lambda+\mu)(\nabla^h\cdot)^\top \nabla^h \cdot \boldsymbol{u} - \mu\Delta_h \boldsymbol{u} =: \boldsymbol{A}\boldsymbol{u}, \tag{6}$$

where Δ_h is the usual seven points discrete vector Laplacian. Note that the approach in [7] is diffusion based (i.e., setting $\lambda = -\mu$ in (2b) and (5)), such that coupling terms vanishes. In contrast to [15], where the strong form given by the product $\mathcal{B}^*\mathcal{B}$ is discretized, we discretize $\mathcal{B}$ and build the discrete product $\boldsymbol{A} := \boldsymbol{B}^\top\boldsymbol{B}$. A discretization of the product $\mathcal{S}_{\boldsymbol{u}}^h \approx \mathcal{B}^*\mathcal{B}$ may not be directly linked to a discrete objective function of the form $\int(\mathcal{B}\boldsymbol{u})^2 d\boldsymbol{x}$. Thus our discretization is a mimetic discretization (see [16]).

The advantage of this approach is that the derivative $\mathcal{S}_{\boldsymbol{u}}^h$ is automatically the derivative of the discrete function $\mathcal{S}^h$. Moreover, boundary conditions are naturally related to those of the operator $\mathcal{B}$.

2.3. Discretizing T and R. If we require a continuously differentiable objective function, we need a continuous image model. For noisy images we use a smoothing B-spline to approximate the images where the smoothing parameter is chosen using the GCV [9, 22]. For data interpolation using B-splines see [25]. Since the grid is regular, we can quickly evaluate the spline coefficients using a cosine transform. The continuous smooth approximation is denoted by T^{spline}.

We are looking for a fast and efficient optimization scheme, and therefore differentiability does play a key role. Thus, as previously discussed in [22], although computationally superior, d-linear image approximations cannot be used, since they are not continuously differentiable. In our implementation we use a cubic B-spline approximation. For further discussion on the effects of higher or lower order B-spline interpolation on the quality of the registration we refer to [22].

Given the staggered grid representation of $\boldsymbol{u}$ we use averaging operators P_j for the transfer to the cell centered positions, and we set

$$\begin{aligned}(P_j u^j)_{\dots,i_j,\dots} &= \frac{1}{2}\Big(u^j_{\dots,i_j+\frac{1}{2},\dots} + u^j_{\dots,i_j-\frac{1}{2},\dots}\Big)\\ T(\boldsymbol{u}) &:= T^{\text{spline}}(x^1 + P_1 u^1, \dots, x^d + P_d u^d);\end{aligned}$$

see [13] for details. We denote the Jacobian of T by

$$T_{\boldsymbol{u}} := \frac{\partial T}{\partial \boldsymbol{u}}(\boldsymbol{u}) = (\operatorname{diag}(P_1^\top \partial_1 T), \dots, \operatorname{diag}(P_d^\top \partial_d T)), \tag{7}$$

where the partial derivatives $\partial_j T$ are evaluated at the spatial positions $(x^1 + P_1 u^1, \dots, x^d + P_d u^d)$. Note that, using a spline approximation for T, $T_{\boldsymbol{u}}$ is a sparse matrix with only four nonzero diagonals.

2.4. Discretizing $\mathcal{D}$. Our discretization of SSD (2a) is straightforward:

$$\boldsymbol{D}(\boldsymbol{u}) := \frac{1}{2}\|T(\boldsymbol{u}) - R\|_2^2 \quad \text{and thus} \quad \boldsymbol{D}_{\boldsymbol{u}}(\boldsymbol{u}) = T_{\boldsymbol{u}}(\boldsymbol{u})^\top (T(\boldsymbol{u}) - R).$$

3. Optimization. The discretized analogue of the image registration problem (1) for fixed α reads as follows:

$$\text{find} \quad \boldsymbol{u} \quad \text{such that} \quad \boldsymbol{J}^\alpha(\boldsymbol{u}) := \boldsymbol{D}(\boldsymbol{u}) + \alpha \boldsymbol{S}(\boldsymbol{u}) = \min. \tag{8}$$

To solve this problem numerically we use an inexact Gauss–Newton method; cf., e.g., [20].

Starting with the initial guess $\boldsymbol{u}$, at each step we compute the gradient $\boldsymbol{J}_{\boldsymbol{u}}^\alpha$ of (8)

$$\boldsymbol{J}_{\boldsymbol{u}}^\alpha = T_{\boldsymbol{u}}(\boldsymbol{u})^\top (T(\boldsymbol{x} + \boldsymbol{u}) - R(\boldsymbol{x})) + \alpha \boldsymbol{A}\boldsymbol{u} \tag{9}$$

and approximately solve the system of linear equations

$$\boldsymbol{H}\delta_{\boldsymbol{u}} = -\boldsymbol{J}_{\boldsymbol{u}}^\alpha, \tag{10}$$

using a multigrid technique which is described in section 4. Here $\boldsymbol{H}$ is an approximation to the Hessian,

$$\boldsymbol{H} = \boldsymbol{M} + \alpha \boldsymbol{A}, \tag{11}$$

and $\boldsymbol{M}$ is an approximation to $T_{\boldsymbol{u}}(\boldsymbol{u})^\top T_{\boldsymbol{u}}(\boldsymbol{u})$. For example, for $d = 3$, we have

$$T_{\boldsymbol{u}}(\boldsymbol{u})^\top T_{\boldsymbol{u}}(\boldsymbol{u}) = \begin{pmatrix} (\partial_1 T)^2 & (\partial_1 T)(\partial_2 T) & (\partial_1 T)(\partial_3 T) \\ (\partial_1 T)(\partial_2 T) & (\partial_2 T)^2 & (\partial_2 T)(\partial_3 T) \\ (\partial_1 T)(\partial_3 T) & (\partial_2 T)(\partial_3 T) & (\partial_3 T)^2 \end{pmatrix}.$$

Although from an optimization standpoint it is better to use the matrix $\boldsymbol{M} = T_{\boldsymbol{u}}(\boldsymbol{u})^\top T_{\boldsymbol{u}}(\boldsymbol{u})$, the usage of this matrix results in a more complicated multigrid solver. As discussed in [18] a simple restriction cannot be used effectively, and an operator-induced interpolation is needed. Such interpolation requires more memory and is slower overall than the simple geometric multigrid scheme. In order to have a simple multigrid scheme, which uses less storage, we approximated this matrix by

$$\boldsymbol{M} = \begin{pmatrix} (\partial_1 T)^2 & 0 & 0 \\ 0 & (\partial_2 T)^2 & 0 \\ 0 & 0 & (\partial_3 T)^2 \end{pmatrix}.$$

Such a choice leads to linear convergence of the optimization scheme even if the problem has a low residual. Nevertheless, it is similar to other Newton multigrid methods [23], since each iteration can be done quickly; then our approach tends to be faster than the full Gauss–Newton approach.

The deformation is updated by $\boldsymbol{u} \leftarrow \boldsymbol{u} + \gamma\delta_{\boldsymbol{u}}$, where γ is chosen according to an Armijo line search, and the process is repeated until convergence, i.e., $\|\boldsymbol{u} - \boldsymbol{u}_{\text{old}}\| \leq \text{tol}$. The algorithm is summarized in Algorithm 1.

ALGORITHM 1. Gauss–Newton method for image registration:
$\boldsymbol{u} \leftarrow \text{GNIR}(\alpha, \boldsymbol{u})$;

```
while true do
  compute J_u^α and H;
  approximately solve H δ_u = −J_u^α;
  if ‖δ_u‖ < tol then
    break;
  end if
  using a line search set u ← u + γδ_u
end while
```

3.1. Continuation in α. As discussed in the introduction, the optimal value of the regularization parameter α is in general unknown a priori. In order to estimate a reasonable α, we follow the strategy suggested in [12]. Starting with a large α_0, we compute a sequence of solutions $\boldsymbol{u}^{\alpha_j}$, where $\alpha_{j+1} < \alpha_j$. This sequence can be viewed as a continuation with respect to the regularization parameter α.

For a large regularization parameter the registration problem is almost quadratic, and therefore its solution requires only a few iterations using a Newton-type optimization technique. Let $j = 0$; starting with α_j, we compute $d_j := \boldsymbol{D}(\boldsymbol{u}^{\alpha_j})$ and expect at least the solution $\boldsymbol{u}^{\alpha_0}$ to under-fit the data. We use $\boldsymbol{u}^{\alpha_j}$ as an excellent starting guess for the computation of the solution of the registration problem for α_{j+1}. We stop if a solution $\boldsymbol{u}^{\alpha_j}$ becomes too irregular. Regularity has to be monitored by a trained professional. Technically, the regularity parameter of our code is given by c. The algorithm is summarized in Algorithm 2.

ALGORITHM 2. Continuation in α: $[\boldsymbol{u}, \alpha] \leftarrow \text{continuation}(\boldsymbol{u}, \alpha, c)$

set $\boldsymbol{u}_0 \leftarrow \boldsymbol{u}$, choose $\eta < 1$;
while true **do**
  solve (1) using Alg. 1: $\boldsymbol{u} \leftarrow \text{GNIR}(\alpha, \boldsymbol{u})$;
  if $\mathcal{D}(\boldsymbol{u}) < c$ **then**
    break;
  end if
  relax the regularization parameter: $\alpha \leftarrow \eta\alpha$;
end while

4. A multigrid solution of the linear systems. The challenging part within the nonlinear optimization is the solution of the linear systems. Each system can be very large and is strongly coupled, but it has a few characteristics which are particularly amenable to multigrid methods. We now discuss a multigrid method for the solution of the linear system. General comments on smoothing, prolongation, restriction, and coarse grid solution are given in [23].

4.1. *h*-ellipticity. To show that the discrete equations (10) are amenable to multigrid methods, we first analyze them by local Fourier analysis (LFA) and show that the system is h-elliptic. It is well known (see, e.g., [23, p. 121–129]) that h-ellipticity is crucial for any multigrid method to be effective. For simplicity, we perform the analysis in two dimensions, but it can be directly generalized into three dimensions.

Recall that the Hessian for the unconstrained optimization problem reads as

$$\boldsymbol{H} = \alpha\boldsymbol{A} + \boldsymbol{M},$$

where $\boldsymbol{M}$ is a nonnegative diagonal matrix. An LFA with freezing coefficients takes the worst-case scenario into account, i.e., $\boldsymbol{M} = \boldsymbol{0}$. Therefore, we need to study the matrix $\boldsymbol{A}$ alone, which is a discretization of the Navier–Lamé operator

$$\begin{aligned}\mathcal{A} = \mathcal{B}^*\mathcal{B} &:= -\mu\Delta - (\lambda+\mu)\nabla\nabla\cdot \\ &= \begin{pmatrix} (2\mu+\lambda)\partial_{x_1,x_1} + \mu\partial_{x_2,x_2} & (\lambda+\mu)\partial_{x_1,x_2} \\ (\lambda+\mu)\partial_{x_1,x_2} & \mu\partial_{x_1,x_1} + (2\mu+\lambda)\partial_{x_2,x_2} \end{pmatrix},\end{aligned}$$

where the so-called Lamé constants λ and μ reflect material properties. Staggered grid discretization leads to the following four stencils for the four parts of the operator (see [23, p. 302] for notation):

$$\left.\begin{aligned} S^{1,1} &= \begin{pmatrix} 0 & -a & 0 \\ -b & 2(a+b) & -b \\ 0 & -a & 0 \end{pmatrix}, \quad S^{1,2} = c\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \\ S^{2,1} &= (S^{1,2})^\top, \qquad\qquad S^{2,2} = (S^{1,1})^\top, \end{aligned}\right\} \tag{12}$$

abbreviating $a = (2\mu+\lambda)$, $b = \mu$, and $c = \lambda+\mu$.

Applying the discretized operators $\boldsymbol{A}_h$ to a grid function $\chi(\boldsymbol{x}, \boldsymbol{\theta}) = e^{\frac{i}{h}\theta^\top \boldsymbol{x}}$, we obtain the symbols of our discretized operators:

$$\widetilde{A}_h(\boldsymbol{\theta}) = \begin{pmatrix} 2aw_1 + 2bw_2 & -4c\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ -4c\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} & 2bw_1 + 2aw_2 \end{pmatrix},$$

where $w_j := 1 - \cos\theta_j = 2\sin^2\frac{\theta_j}{2}$. Its determinant is given by

$$\begin{aligned} d(\boldsymbol{\theta}) = \widetilde{d}(\boldsymbol{w}) &= 4abw_1^2 + 4abw_2^2 + 4(a^2+b^2)w_1w_2 - 4c^2w_1w_2 \\ &= 4ab(w_1+w_2)^2. \end{aligned} \tag{13}$$

THEOREM 4.1. *The h-ellipticity measure of the staggered grid discretizations* (12) *is*

$$E_h(\boldsymbol{A}_h) = \frac{1}{4},$$

and in particular it is independent *of the Lamé constants.*

Proof. Following [23, section 8.3.2],

$$E_h(\boldsymbol{A}_h) := \frac{\min\{|\det \widetilde{A}_h| :\ \theta \in T^{\text{high}}\}}{\max\{|\det \widetilde{A}_h| :\ -\pi < \theta_j \le \pi\}}.$$

From (13), we see that the min and max are independent of λ and μ:

$$\begin{aligned} \max\{d(\boldsymbol{\theta}) :\ -\pi < \theta_j \le \pi\} &= \max\{\widetilde{d}(\boldsymbol{w}) :\ 0 \le w_j \le 1\} \\ &= \widetilde{d}(1,1) = 16ab = 16\mu(2\mu+\lambda), \\ \min\{d(\boldsymbol{\theta}) :\ \theta \in T^{\text{high}}\} &= \min\left\{\widetilde{d}(\boldsymbol{w}) :\ \frac{1}{2} \le |w_j| \le 1\right\} \\ &= \widetilde{d}\left(\frac{1}{2},\frac{1}{2}\right) = 4ab = 4\mu(2\mu+\lambda). \end{aligned}$$

∎

Remark 4.2. The above theorem is remarkable because as the ratio λ/μ becomes larger, the system becomes more ill-conditioned. Nevertheless, the theorem suggests that an appropriate multigrid implementation works well regardless of that ratio. Note also that other iterative techniques (such as conjugate gradient) are very sensitive to this ratio while our multigrid method is not.

4.2. Smoothing. The h-ellipticity of our discretization (independent of choices of the Lamé constants λ and μ) guaranties that, for example, a Kaczmarz relaxation has smoothing properties. Here we have implemented a damped Jacobi smoother with damping parameter of 2/3. Since we want our code to run on modest computational architecture and use only a small amount of memory, our implementation does not generate the matrix and only matrix vector products are calculated.

4.3. Prolongation and restriction. To fulfill the relation between the order of the differential operator and the sum of the orders of prolongation and restriction, we use linear interpolation for prolongation and its transpose for the restriction. This is similar to prolongation and restriction used in computational fluid dynamics. For an elaborate discussion on these operations on a staggered grid see [26, p. 253].

4.4. Coarse grid operator. Although it is possible to use a Galerkin coarse grid operator for building the coarse grid system $\boldsymbol{A}_H$, a straightforward implementation requires either additional memory or additional computations. For problems with uniform and structured grids, one can use rediscretization to get a faster implementation. However, since rediscretization cannot be applied to the diagonal matrix $\boldsymbol{M}_h$, we choose the following approach leading to a diagonal $\boldsymbol{M}_H$. In the first stage we

extract the diagonal entries of $\boldsymbol{M}_h$; this corresponds to a vector with the same dimensions of $\boldsymbol{u}_h$. Second, we generate the diagonal matrix $\boldsymbol{M}_H$ by restricting this vector using the same restriction operator designed for $\boldsymbol{u}_h$, and set the diagonal elements of $\boldsymbol{M}_H$ to the restricted values.

4.5. Multigrid cycles. The above components are linked together to a V- or a W-cycle. In our implementation we use only a single V-cycle for each nonlinear iteration. As we show in the numerical experiments, a $V(3,1)$-cycle is very effective and decreases the residual in one or two orders of magnitude.

5. Multilevel continuation. Image registration can be highly nonlinear, and therefore, one may require many iterations to achieve a solution. An important method to save computational work and to deal with nonlinearities is to use a multilevel continuation. Multilevel continuation is well established for optimization problems and systems of nonlinear equations; see, e.g., [1, 2]. However, in image registration it has an additional advantage. Similar to [10, 4], we use the multilevel approach to efficiently identify the relevant range of α's. Thus, our multilevel approach uses continuation in both the grid and the regularization parameter. Note that even for approaches where the images may not be smooth, the desired solution $\boldsymbol{u}^\alpha$ is smooth. Therefore, multilevel methods provide an effective tool for the continuation problem.

Given an initial upper bound C for the image distance, using a secant method, we compute a parameter α_C, such that

$$\mathcal{D}(\boldsymbol{u}^{\alpha_C}) \approx C. \tag{14}$$

After α_C is computed, we use Algorithm 2 on the fine grid to obtain a sequence of solution $\boldsymbol{u}^\alpha$, where $c \le \mathcal{D}(\boldsymbol{u}^\alpha) \le C$. We summarize the multilevel algorithm in Algorithm 3.

ALGORITHM 3. Multilevel image registration: $\boldsymbol{u} \leftarrow$ MLIR

```
choose u on the coarsest grid;
while true do
  solve the registration problem u = GNIR(u, α); cf. Alg. 1;
  if |D(u) − C| ≥ tol then
    adjust α using a secant method;
  end if
  if on finest grid then
    return;
  end if
  prolongate u to the finer grid: u ← prolongate(u);
end while
```

6. Numerical examples. In this section we perform numerical experiments that demonstrate the effectiveness of our algorithm. We present a registration of three-dimensional MRI scans of a human knee (images provided by Thomas Netsch, Philips Medical Systems, Hamburg, Germany). The problem is known to be hard because large nonlinear deformations are needed in order to perform the registration; see also [17].

As is apparent from Figure 2, the reference shows the knee in an almost straight position, and the template shows the knee in a bent position. The deformation is thus

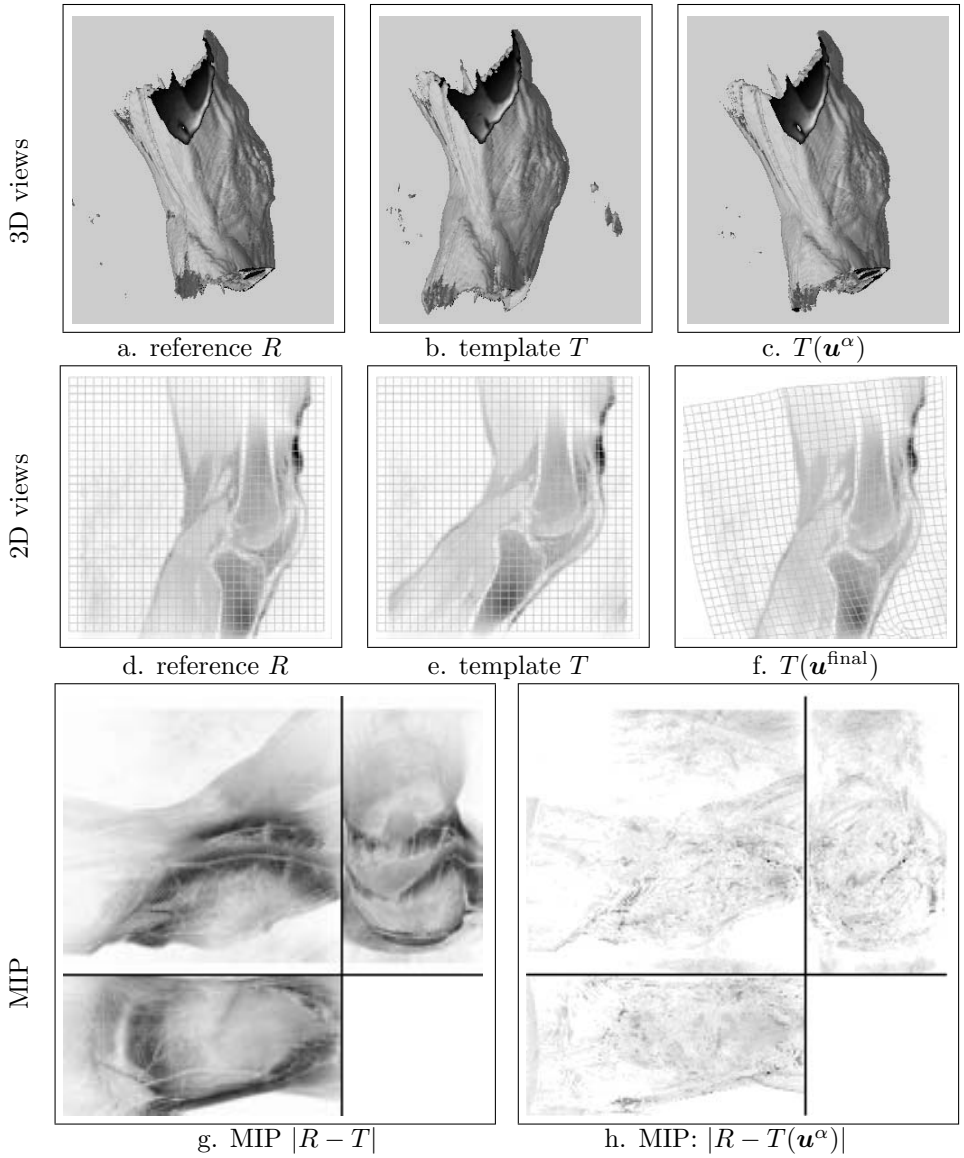

FIG. 2. *Registration results for two* $128\times128\times64$ *MRI scans of a human knee; three-dimensional views of* R (a), T (b), *and* $T(\boldsymbol{u}^{\alpha})$ (c)*; the generic slice* 40 *with grid for* R (d), T (e), *and* $T(\boldsymbol{u}^{\alpha})$ (f)*; maximum intensity projections for* $|R-T|$ (g) *and* $|R-T(\boldsymbol{u}^{\alpha})|$ (h).

expected to be highly nonlinear. To clarify the example, we divide this section into two parts. In the first part we go over the computational details, while in the second we discuss the results from an imaging point of view.

As common to many other registration algorithms we use a two-phase registration (see [19]). In the first phase a multilevel linear preregistration is done, followed by a second-phase multilevel nonlinear registration. The linear registration managed to reduce the image distance to 50.1%, and the rest was done using our nonlinear registration.

TABLE 1
Reductions in the SSD (compared with the initial SSD in %), and iterations (in #) for a decreasing α (image size $128 \times 128 \times 64$).

α	Iter	SSD
$8.5 \cdot 10^{-4}$	1	15.6%
$4.2 \cdot 10^{-4}$	1	12.2%
$2.1 \cdot 10^{-4}$	2	10.9%
$1.0 \cdot 10^{-4}$	2	10.1%

All computations are performed using MATLAB 6.1 on a 2.2GHz laptop with 1GB of RAM. The overall time for a registration ($\alpha = 10^{-3}$) was about 118 minutes, including the computation for the B-spline coefficients ($\approx$ 11 minutes). Since the signal-to-noise ratio in the images is high, for the computation of the B-spline we assume that the images do not contain any noise (thus the B-splines interpolate the images). In our computations, about one third of the time is needed for the linear part, and about two thirds is needed for the nonlinear part. Within the nonlinear part, 66% are needed for the computation of $T(\boldsymbol{u})$ and $T_{\boldsymbol{u}}(\boldsymbol{u})$, and 24% are needed to solve the linear system. An overview with respect to levels is presented in Table 2.

6.1. Computational details. We present three types of experiments. In our first example, we run the algorithm precisely as described in this paper (Algorithm 3); i.e., starting on the coarse grid we compute an initial α and adjust it by using the continuation techniques described in section 5. In order to present comparable multigrid convergence rates, we also present results for fixed values of α in our second example. Finally, in the third set of experiments we test the sensitivity of our multigrid algorithm to the elasticity parameters λ and μ. Although all image registration algorithms known to us fix this ratio at roughly 1, we experiment with a high ratio which challenges our multigrid method.

6.1.1. Continuation in grid and $\boldsymbol{\alpha}$. In our first numerical experiment we set $\mu = \lambda = 1$ and the constant C in Algorithm 3 to 0.2. Using Algorithm 3 we solve the problem, and this yields α which is $1.71 \cdot 10^{-3}$ (performance and results for $\alpha = 1.71 \cdot 10^{-3}$ and $\alpha = 10^{-3}$ are indistinguishable).

We then follow Algorithm 2, gradually reducing the regularization parameter to 10^{-4} by solving the problem for 4 different α's. For each regularization parameter we use the solution for the previous α as described in Algorithm 2. The number of iterations for each α and the SSD is recorded in Table 1. We observe both the images and the distorted grid for each solution. We stop the iteration at $1.0 \cdot 10^{-4}$ because we observe that the grid is highly distorted. The solution converges quickly (1 or 2 steps) for each new regularization parameter when starting from the solution of the previous regularization parameter. This demonstrates the importance of continuation in α for the robustness of the algorithm.

6.1.2. Sensitivity of the algorithm to the regularization parameter. In our second experiment we test the effect of different regularization parameters on the efficiency of our algorithm. We run our code with the (fixed) regularization parameters $\alpha = 10^{-2}, 10^{-3}, 10^{-4}$. The results of these experiment are documented in Table 2.

As common to many inverse problems, convergence of the optimization algorithm slows down as the regularization parameter decreases (see [12]). This implies that starting with an arbitrary guess on the fine grid may result in many more expensive iterations. Nevertheless, using a multilevel continuation approach, we are able to

TABLE 2
Reductions in the SSD (compared with the initial SSD in %), and iterations (in #) for the multilevel elastic registration; image size is $n^3/2$.

Level	n	Iterations			SSD		
		$\alpha = 10^{-4}$	$\alpha = 10^{-3}$	$\alpha = 10^{-2}$	$\alpha = 10^{-4}$	$\alpha = 10^{-3}$	$\alpha = 10^{-2}$
5	8	21	13	10	26.1%	17.2%	11.3%
4	16	18	16	12	25.9%	17.8%	10.9%
3	32	12	9	7	25.5%	17.9%	10.7%
2	64	9	8	6	25.3%	17.2%	10.2%
1	128	7	6	5	25.3%	17.2%	10.1%

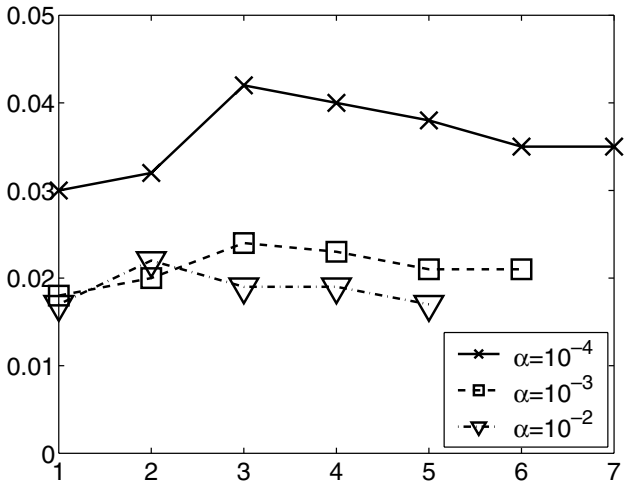

FIG. 3. *Convergence rate for each linear iteration versus iteration for $\alpha = 10^{-2}, 10^{-3}, 10^{-4}$; see also Table 2.*

quickly converge on the finest grid because we have a good starting guess which is obtained on coarser grids. This demonstrates the importance of multilevel continuation.

To observe the effectiveness of our multigrid algorithm solver we record the reduction of the residual for each of the linearized Gauss–Newton iterations on the finest grid. Note that for each iteration we have a different approximation to the Hessian characterized by the different $\boldsymbol{M}$'s. In our experiment, the multigrid solver was very effective, and after only a single $V(3,1)$-cycle our residual is decreased by approximately two orders of magnitudes. The convergence history is plotted in Figure 3.

The convergence properties of the linear iteration was only mildly influenced by the decreasing regularization parameter. This observation is similar to other multigrid methods for inverse problems [5].

6.1.3. Sensitivity of the algorithm to the ratio μ/λ. In the previous experiments, common to many other image registration algorithms [15], we used the ratio $\mu/\lambda = 1$. However, as Theorem 1 shows, the h-ellipticity of our discretized system is 0.25 independent of this choice, which implies in particular that the multigrid iteration should not be affected by this ratio. To put this theory to a test, we vary the ratio μ/λ, starting at 10 to 10^{-3}. Note that for the latter, the regularization operator is close to $\| \nabla \cdot \boldsymbol{u}\|^2$, which is the incompressible limit. However, we emphasize that

TABLE 3
Number of iterations and average residual reduction for different ratios μ/λ.

Ratio	10^1	10^0	10^{-1}	10^{-2}	10^{-3}
Iterations	18	17	16	16	17
Reduction	$2.3 \cdot 10^{-2}$	$2.1 \cdot 10^{-2}$	$2.4 \cdot 10^{-2}$	$2.8 \cdot 10^{-2}$	$2.9 \cdot 10^{-2}$

such a large ratio is meaningless for the purpose of image registration since it results in very irregular grids. From an imaging point of view, only the ratio $\mu/\lambda = 1$ leads to a reasonable displacement (judging by eye inspection). Our experiments are done solely to test our multigrid method.

For these experiments we set $\alpha = 10^{-3}$, and in order to have a long enough iteration history, we run the inversion process on the fine grid alone. For each nonlinear (Gauss–Newton) iteration we perform a single V(3,1)-cycle. For each cycle we record the reduction of the residual. At the end of each registration process we average the reduction of the residuals and record it as the *average residual reduction*. The results of these experiments are summarized in Table 3.

In accordance with the theory, our experiments demonstrate that the multigrid method is in fact not very sensitive the ratio μ/λ. In fact, although the condition number for the system which results from $\mu/\lambda = 10^{-3}$ is roughly 10^3 times the condition number of the system which results from $\mu/\lambda = 1$, the reduction in the residuals (and therefore the convergence) is very minimally influenced. Note that this property is unique to multigrid methods, and a naively preconditioned conjugate gradient method would not have had such properties.

6.2. Qualitative analysis of the images. In Figure 2 we show the results for $\alpha = 10^{-3}$. As is apparent from Figure 2(c), the overall registration is reasonable. We also analyze the results in terms of a pairwise comparison of image slices. A two-dimensional view of the generic slice 40 of a total of 64 is shown in Figure 2(d), (e), (f). We add a regular grid to R and T and the deformed grid to $T(\boldsymbol{u}^\alpha)$. Though highly nonlinear, the deformation appears to be sufficiently smooth, as can be seen from the deformed grid. Finally, Figure 2 also shows maximum intensity projections of the distances $|R - T|$ and $|R - T(\boldsymbol{u}^\alpha)|$, respectively.

Running our scheme with a regularization parameter $\alpha = 10^{-2}$ leads to visually indistinguishable results but to a much higher distance (reduction to 25.3% instead of 17.2% for $\alpha = 10^{-3}$). On the other hand, choosing $\alpha \leq 10^{-4}$ results in nonphysical deformations which are visible on the deformed grid. Therefore, it is easy to derive a lower bound for α. However, choosing an "optimal" regularization parameter α is a delicate matter since it requires a quantification of "physically meaningful." We do not believe in an automatic detection and therefore leave the final choice to an expert.

7. Summary. In this paper we have developed a multilevel registration scheme for image registration based on a variational formulation. The scheme is based on a multilevel inexact Gauss–Newton scheme with a multigrid solver for the linearized systems. We make extensive use of numerical optimization techniques and therefore pay special attention to differentiability issues. We use smoothing B-splines to compute the solution to the forward problem and staggered grids to discretize the transformation. We prove that the discretization is h-elliptic and therefore is amenable to multigrid methods. The numerical experiments support our theoretical prediction and show that we can solve large scale problems using computational tools like MATLAB.

REFERENCES

[1] J. C. Alexander and J. A. Yorke, *The homotopy continuation method: Numerically implementable topological procedures*, Trans. Amer. Math. Soc., 242 (1978), pp. 271–284.

[2] E. Allgower and K. Georg, *Numerical Continuation Methods*, Springer-Verlag, Berlin, 1990.

[3] D. N. Arnold and R. Winther, *Mixed finite elements for elasticity*, Numer. Math., 42 (2002), pp. 401–419.

[4] U. Ascher and E. Haber, *Grid refinement and scaling for distributed parameter estimation problems*, Inverse Problems, 17 (2001), pp. 571–590.

[5] U. Ascher and E. Haber, *A multigrid method for distributed parameter estimation problems*, Electron. Trans. Numer. Anal., 15 (2003), pp. 1–15.

[6] L. G. Brown, *A survey of image registration techniques*, ACM Computing Surveys, 24 (1992), pp. 325–376.

[7] U. Clarenz, M. Droske, and M. Rumpf, *Towards fast non-rigid registration*, in Inverse Problems, Image Analysis and Medical Imaging, Contemp. Math. 313, M. Z. Nashed and O. Scherzes, eds., AMS, Providence, RI, 2002, pp. 67–84.

[8] C. A. J. Fletcher, *Computational Techniques for Fluid Dynamics*, Vol. II, Springer-Verlag, Berlin, 1988.

[9] G. Golub, M. Heath, and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.

[10] E. Haber, *A multilevel, level-set method for optimizing eigenvalues in shape design problems*, J. Comput. Phys., 98 (2004), pp. 518–534.

[11] E. Haber and U. M. Ascher, *Fast finite volume simulation of 3D electromagnetic problems with highly discontinuous coefficients*, SIAM J. Sci. Comput., 22 (2001), pp. 1943–1961.

[12] E. Haber, U. Ascher, and D. Oldenburg, *On optimization techniques for solving nonlinear inverse Problems*, Inverse Problems, 16 (2000), pp. 1263–1280.

[13] E. Haber and J. Modersitzki, *Numerical solutions of volume preserving image registration*, Inverse Problems, 20 (2004), pp. 1621–1638.

[14] P. C. Hansen, *Rank-Deficient and Ill-Posed Problems: Numerical Aspects of Linear Inversion*, SIAM, Philadelphia, 1997.

[15] S. Henn and K. Witsch, *Iterative multigrid regularization techniques for image matching*, SIAM J. Sci. Comput., 23 (2001), pp. 1077–1093.

[16] J. M. Hyman and M. Shashkov, *The orthogonal decomposition theorems for mimetic finite difference methods*, SIAM J. Numer. Anal., 36 (1999), pp. 788–818.

[17] S. Kabus, T. Netsch, B. Fischer, and J. Modersitzki, *B-spline registration of 3d images with Levenberg-Marquardt optimization*, in Proceedings of the SPIE 2004, Medical Imaging, San Diego, 2004.

[18] E. M. Kalmoun and U. Rüde, *A Variational Multigrid for Computing the Optical Flow*, Tech. report, Department of Computer Science, Friedrich-Alexander University, Erlangen-Nuremberg, 2003.

[19] J. Modersitzki, *Numerical Methods for Image Registration*, Oxford University Press, New York, 2004.

[20] J. Nocedal and S. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[21] R. L. Parker, *Geophysical Inverse Theory*, Princeton University Press, Princeton, NJ, 1994.

[22] P. Thevenaz and M. Unser, *Optimization of mutual information for multiresolution image registration*, IEEE Trans. Image Process., 9 (2000), pp. 2083–2089.

[23] U. Trottenberg, C. Oosterlee, and A. Schuller, *Multigrid*, Academic Press, San Diego, 2001.

[24] C. R. Vogel, *Computational Methods for Inverse Problem*, SIAM, Philadelphia, 2002.

[25] G. Wahba, *Spline Models for Observational Data*, SIAM, Philadelphia, 1990.

[26] P. Wesseling, *An Introduction to Multigrid Methods*, John Wiley & Sons, Chichester, UK, 1992.

[27] K. S. Yee, *Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media*, IEEE Trans. Antennas and Propagation, 14 (1966), pp. 302–307.