

A NONLINEAR VARIATIONAL PROBLEM FOR IMAGE MATCHING*

YALI AMIT[†]

Abstract. Minimizing a nonlinear functional is presented as a way of obtaining a planar mapping that matches two similar images. A smoothing term is added to the nonlinear functional to penalize discontinuous and irregular solutions. One option for the smoothing term is a quadratic form generated by a linear differential operator. The functional is then minimized using the Fourier representation of the planar mapping. With this representation the quadratic form is diagonalized. Another option is a quadratic form generated via a basis of compactly supported wavelets. In both cases, a natural approximation scheme is described. Both quadratic forms are shown to impose the same smoothing. However, in terms of the finite dimensional approximations, it is easier to accommodate local deformations using the wavelet basis.

Key words. image matching, movement compensation, nonlinear variational problem, spectral methods, wavelets

AMS subject classifications. 68V10, 65M70, 49N60

1. Introduction. Let F and G be smooth functions on the two-dimensional unit square I^2 , and let $\phi(x)$ be a mapping of the unit square into itself, such that $G(x) = F(\phi(x))$. It is clear that if F and G have the same range, many such mappings exist, most of which would be highly discontinuous and degenerate and not very interesting. However, if the mapping $\phi(x)$ is a smooth diffeomorphism of the unit square onto itself, then the extremal points of F would be mapped via ϕ^{-1} onto the extremal points of G , level curves would be mapped onto level curves, etc. Heuristically speaking, the graphs of F and G considered as surfaces would have similar topographies. Conversely, if F and G have similar topographies, then it should be possible to find a smooth and locally nondegenerate mapping ϕ such that $F(\phi(x))$ is close to G in some sense.

To illustrate this idea, consider the images in Fig. 1, which are x-rays of two different hands. If we consider the images as some smooth function sampled at the points of the pixel lattice, we obtain two functions that indeed have very similar topographies. This would be the case with any two images of some fixed organ of the body of two different patients, or of the same patient obtained at different times, provided that these images came from the same type of imaging device. Consequently, there should exist a smooth and locally nondegenerate mapping ϕ that transforms one image, called the *template*, into the other image, called the *data*, via composition. The mapping ϕ would automatically match between the corresponding parts of the two images. See, for example, Fig. 1 where the various parts of the hand such as the tips of the fingers or the joints are correctly matched.

One of the first attempts dealing with the issue of image matching can be traced to Horn and Schnuck [5] and Huang and Tsai [6] in the context of optical flow and movement compensation calculations for sequences of images. These ideas were further developed by Nagel [10] and Terzopoulos [11]. In Bajcy and Kovacic [2] these ideas were applied to the issue of matching medical images of similar organs, such as MRI images of the brain. Here the matching is not intended to calculate movement, but to automate the analysis of medical images. This second problem is also more difficult in that large deformations may occur, as opposed to relatively small deformations in image sequences.

*Received by the editors April 1, 1991; accepted for publication (in revised form) March 1, 1993. This research was supported in part by Office of Naval Research grant N00014-88-K-0289 and Army Research Office grant DAAL03-90-G-0033.

[†]Department of Statistics, University of Chicago, Chicago, Illinois 60637. This work was done while the author was at Brown University, Division of Applied Mathematics, Providence, Rhode Island 02912 (amit@galton.uchicago.edu).

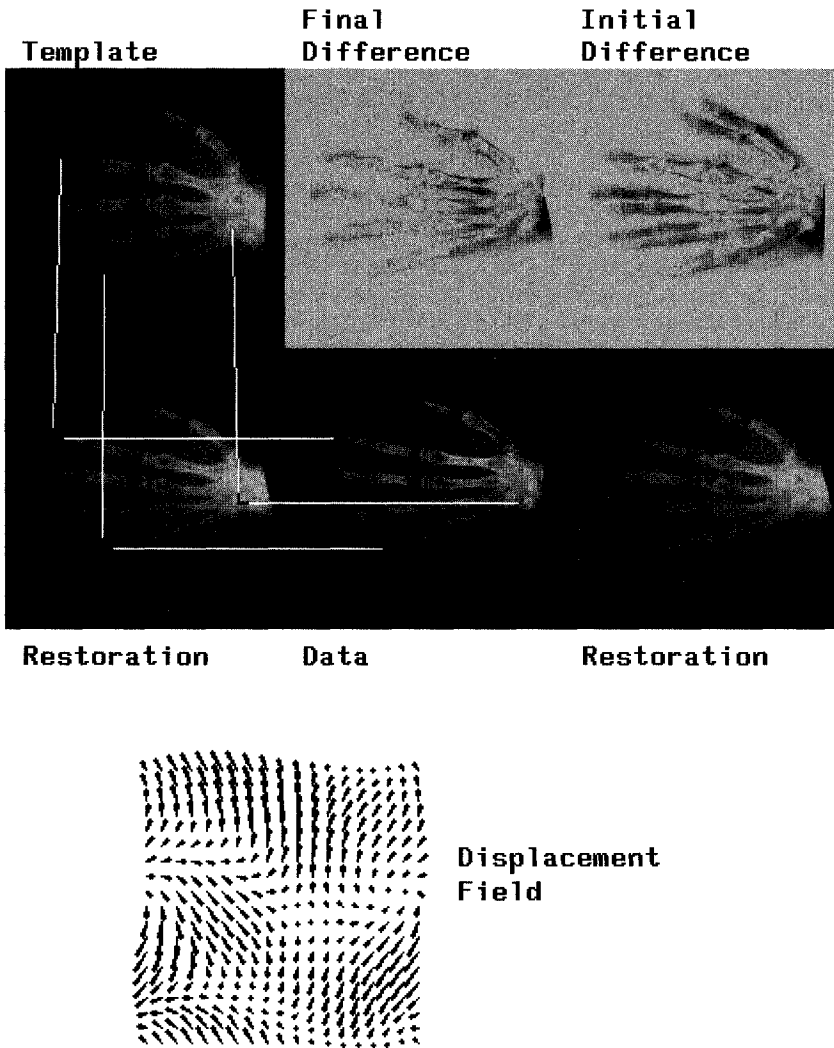


FIG. 1. The template corresponds to the function F and the data to the function G . The restoration was done using the Fourier method. The white lines show the matching induced by the displacement field. A point x in the data is connected via the restoration to the point $x + \bar{U}(x)$ from which it obtained its grey level value. The final difference image shows $1 - |F(x + \bar{U}(x)) - G(x)|$ and the initial difference image shows $1 - |F(x) - G(x)|$.

The question is how to find the mapping ϕ . We have addressed this problem by minimizing the following functional:

$$(1) \quad I(U) = \frac{1}{2} \int_{T^2} |F(x + U(x)) - G(x)|^2 dx,$$

where $U(x) = \phi(x) - x$ is the displacement field and T^2 is the unit torus. Minimizing I over some set of vector fields provides a mapping $\phi(x) = x + U(x)$ of the torus into itself, such that $F \circ \phi(x)$ is close in the mean square norm to G .

It should be noted that the periodic domain is chosen for the sake of notational and computational convenience. It takes care of the problem of what to do when $x + U(x)$ is not

in I^2 . Another possibility is that F is defined on some large domain D that includes I^2 . Then one would want to minimize over mappings from I^2 into D .

To rule out the discontinuous and irregular solutions to this minimization problem, it is possible to introduce a smoothing or regularizing term, thus obtaining a new functional

$$(2) \quad J(U) = \alpha \mathcal{E}(U, U) + \frac{1}{2} \int |F(x + U(x)) - G(x)|^2 dx,$$

where \mathcal{E} is a bilinear form penalizing nonsmooth functions. In all our applications, \mathcal{E} was taken to be a Hilbertian norm equivalent to one of the Sobolev norms. Since the domain under consideration is two dimensional, taking the Sobolev norm to be of order greater or equal to two will ensure that the solutions are continuous. Higher-order Sobolev norms will, of course, introduce additional smoothness. This approach was first described in a statistical setting by Amit, Grenander, and Piccioni [1].

The functional J is nonlinear and may have many global and local minima. In the sequel we will be interested mainly in finding local minima of J close to the initial point $U(x) \equiv 0$ that corresponds to the identity map. The nondegeneracy of the mapping $\hat{\phi}$ generated by a local minimizer \hat{U} is then ensured by the fact that it is close to the identity map so that its Jacobian is nonzero at most points.

There are several major differences between the work mentioned above and the approach presented here. First, we do not use a data term derived from intensity conservation assumptions originally suggested by Horn and Schnuck [5], which is equivalent to linearizing the functional I as described at the end of §2. It appears that the linearized problem will not capture larger deformations (see Fig. 2).

Second, the solution of the variational problem is obtained by parametrizing the unknown function in terms of its coefficients with respect to either the Fourier basis or some wavelet basis, thus allowing for a coarse-to-fine or multiresolution approach. This was indeed suggested in [11] using multigrid techniques, which may be appropriate for the linearized equations that have a unique solution and are known to be efficiently soluble using multigrid techniques. However, given that the nonlinear functional is to be used, and that this nonlinear functional is not convex, it is not clear how well the classical multigrid approach will perform. The coarse level displacement is calculated using only the information of a smoothed version of the data on that same coarse grid, and there is some risk of information being lost. Moreover, when moving to finer grids, a bilinear interpolation is used that may not be smooth enough and that may introduce unnatural deformations.

Setting the problem in terms of an orthonormal basis directly incorporates interpolation through the basis functions. The smoothing operator is automatically written in diagonal form in terms of the basis chosen. Thus using the description in terms of a basis expansion, and solving first for low-frequency coefficients, gradually increasing the number can be thought of as a multigrid method translated onto the finest grid. Although some computational speed is lost, the advantage is that all the data is used to drive the algorithm.

The level of smoothness versus locality can be controlled by the choice of wavelet basis. Since the problem at hand is not really governed by physical fluid dynamical or elasticity laws, there is no special advantage in using the Laplacian as a smoothing operator. The existence of fast transforms for these bases makes the algorithm computationally feasible.

In §2 the smoothing term \mathcal{E} is set to be a quadratic form generated by a linear differential operator. The approximations are then described together with minimization procedure. The basic idea is to diagonalize the differential operator using the Fourier basis and to solve the problem in the spectral domain.

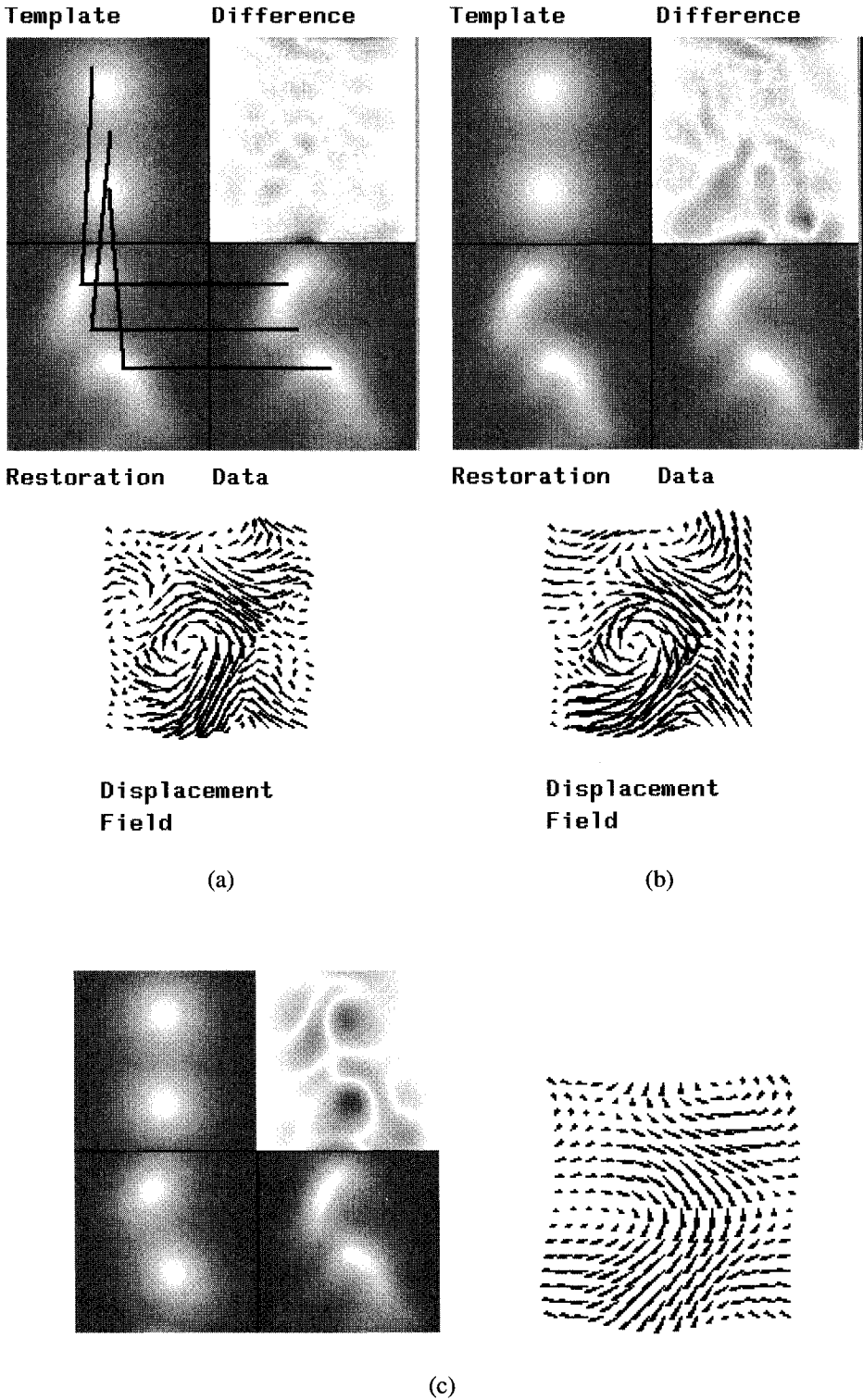


FIG. 2. A transformation constructed with the Fourier basis was initially applied to the template to generate the data. (a) was done using the Fourier basis. (b) was done using the wavelet basis. (c) was done using the linearized equations.

In §3 an alternative smoothing term is suggested. This time \mathcal{E} is directly given in a diagonal form using a wavelet basis instead of the Fourier basis. The eigenvalues are set so as to ensure the same type of smoothing.

In §4 the experiments are described, and the performance of the two approaches is compared.

2. A nonlinear partial differential equation. Consider the bilinear form

$$B(f, g) = \frac{1}{2} \int_{T^2} (\Delta + \varepsilon)^2 f(x) \cdot g(x) dx,$$

where Δ denotes the Laplacian with periodic boundary conditions. This bilinear form defines a Hilbertian norm equivalent to the standard Sobolev norm on $H = H^2(T^2)$. Set $\varepsilon(U, U) = B(U^{(1)}, U^{(1)}) + B(U^{(2)}, U^{(2)})$ in (2). With this choice of the bilinear form in the functional J , the Euler equation for the minimizer is the following nonlinear partial differential equation (PDE):

$$\begin{aligned} \alpha(\Delta + \varepsilon)^2 U^{(1)} &= (G(x) - F(x + U(x))) \frac{\partial F}{\partial x_1}(x + U(x)), \\ \alpha(\Delta + \varepsilon)^2 U^{(2)} &= (G(x) - F(x + U(x))) \frac{\partial F}{\partial x_2}(x + U(x)). \end{aligned}$$

The parameter α determines the relative weight of the regularizing term. Since the issue of the choice of α is not addressed here, we set $\alpha = 1$. This is the parameter used in the experiments as well.

For the purpose of numerical solutions, it is, of course, necessary to find finite-dimensional approximations to the functional J . Since the spectral decomposition of Δ is known, it is convenient to write J in the spectral domain and then approximate it coordinatewise. Let λ_{kl}, ϕ_{kl} denote the eigenvalues and eigenvectors of Δ , then

$$\lambda_{kl} = (2\pi)^2(k^2 + l^2) + \varepsilon \quad \text{and} \quad \phi_{kl}(x_1, x_2) = e^{2\pi i(kx_1 + lx_2)},$$

and the functional J can be rewritten as

$$\begin{aligned} J(U) &= \sum_{k,l=-\infty}^{\infty} \lambda_{kl}^2 ((u_{kl}^{(1)})^2 + (u_{kl}^{(2)})^2) \\ &\quad + \int \left[F\left(x_1 + \sum u_{kl}^{(1)} \phi_{kl}(x), x_2 + \sum u_{kl}^{(2)} \phi_{kl}(x)\right) - G(x) \right]^2 dx \\ &= \Gamma(u^{(1)}, u^{(2)}) + q(u^{(1)}, u^{(2)}), \end{aligned}$$

where $u_{kl}^{(i)} = \int U^{(i)}(x) \phi_{kl}(x) dx$. The vector $(u^{(1)}, u^{(2)}) \in \ell_2 \times \ell_2$, is simply the coordinate vector of $(U^{(1)}, U^{(2)})$ with respect to the basis ϕ_{kl} of $L_2(T^2)$. We write $(u^{(1)}, u^{(2)}) = \pi(U^{(1)}, U^{(2)})$. Note that $q(u^{(1)}, u^{(2)}) = I(U^{(1)}, U^{(2)})$ with I as in (1).

The finite dimensional approximations of the functional are obtained by taking the sums in the linear term and those in the integrand between $-(N - 1)$ and N . The approximation is therefore obtained by restricting the argument of J to the space $H_N \times H_N$, where $H_N = \text{span}\{\phi_{kl}\}_{k,l=-(N-1)}^N$. The dimension of the approximate space H_N is $(2N)^2$ and $\pi(H_N \times H_N) = R^{(2N)^2} \times R^{(2N)^2}$.

Let J_N denote the approximate functional on $H_N \times H_N$. Each of the finite dimensional functionals is positive and continuous. Moreover, $J_N(U) \rightarrow \infty$ as $U \rightarrow \infty$ and therefore has at least one global minimum. Let S_N be the set of global minima of J_N .

Downloaded 04/22/14 to 128.135.149.17. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

THEOREM. *Let $U_N \in S_N$ for $N = 1, 2, \dots$. Then U_N has a convergent subsequence in $H \times H$, which converges to a global minimum of J .*

Proof. Consider $H \times H$ with the Hilbertian norm defined by $\mathcal{E}(\cdot, \cdot)$. Since $H \times H$ is compactly imbedded in $C(T^2) \times C(T^2)$, $|U_N - U|_{H \times H} \rightarrow 0$ implies uniform convergence that, in turn, implies that $G(x + U_N(x)) \rightarrow G(x + U(x))$, as $N \rightarrow \infty$. Since G and F are bounded, it follows from the dominated convergence theorem that $I(U_N) \rightarrow I(U)$. Clearly, $\mathcal{E}(U_N, U_N) \rightarrow \mathcal{E}(U, U)$, so that J is continuous in H .

Now $J_N(U_N)$ is a positive monotonically decreasing sequence so that it converges to some value L . Moreover, since U_N is the global minimum of J in $H_N \times H_N$, and since $\cup_{N=1}^\infty (H_N \times H_N)$ is dense in $H \times H$, it follows from the continuity of J that $L = \inf_{U \in H \times H} J(U)$.

Since $|U_N|_{H \times H} = \mathcal{E}(U_N, U_N) < J_N(U_N) < J_1(U_1)$ for all $N = 1, 2, \dots$ the sequence U_N is weakly compact. Let U_{N_k} be a subsequence that converges weakly to some $U \in H \times H$. Again, since H is compactly embedded in $C(T^2)$, U_{N_k} converges in the uniform norm to U as $k \rightarrow \infty$. As above, this implies that $I(U_{N_k}) \rightarrow I(U)$. Together with the fact that $J(U_{N_k})$ converges, we conclude that $\mathcal{E}(U_{N_k}, U_{N_k})$ converges to $L - I(U)$ as $k \rightarrow \infty$.

Now since the norm function is lower semicontinuous in the weak topology, we have $\mathcal{E}(U, U) \leq L - I(U)$, and since L is the infimum, we have $\mathcal{E}(U, U) + I(U) = L$ and $\mathcal{E}(U, U) = \lim_{k \rightarrow \infty} \mathcal{E}(U_{N_k}, U_{N_k})$. This, together with the weak convergence of U_{N_k} to U , implies strong convergence in H . Finally, since $J(U) = L$, it is a global minimum of J . \square

Observe that by using the same arguments as above, it is possible to show that the set of global minima of J is compact in H .

Practical considerations. In practice, the template and the data are images and are given only on a discrete pixel lattice of equally spaced points $x_{\alpha\beta}$, $\alpha, \beta = 0, \dots, L - 1$ in I^2 . The unknown displacement field U is given in terms of its array of values at the points of the lattice, which will also be denoted by U . We write $U_{\alpha\beta} = U(x_{\alpha\beta})$ for all $\alpha, \beta = 0, \dots, L - 1$. The approximate functional J_N now has the form

$$(3) \quad J_N(U) = \sum_{k,l=-(N-1)}^N \lambda_{kl} ((u_{kl}^{(1)})^2 + (u_{kl}^{(2)})^2) + \frac{1}{L^2} \sum_{\alpha,\beta=0}^{L-1} [F(x_{\alpha\beta} + U_{\alpha\beta}) - G(x_{\alpha\beta})]^2.$$

The array $u_{k,l}$, $k, l = -(N - 1), \dots, N$ is the discrete Fourier transform of the array U , i.e.,

$$u_{kl}^{(i)} = \frac{1}{L^2} \sum_{\alpha,\beta=0}^{L-1} U_{\alpha\beta}^{(i)} \phi_{kl}(x_{\alpha\beta}),$$

and we write $u^{(i)} = \pi(U^{(i)})$. Since F is actually given only on the lattice, and $x_{\alpha\beta} + U_{\alpha\beta}$ may not lie on the lattice, it is possible either to truncate to the nearest point or to use a linear interpolation between the four nearest points. The resolution of the pixel lattice is the finest, so that these corrections are negligible.

We have not tried to find the global minimum of the approximating functional. Instead, we have done gradient descent starting at initial point zero. In other words, the following ordinary differential equation (ODE) in $R^{(2N)^2} \times R^{(2N)^2}$ was solved.

$$(4) \quad \begin{aligned} \frac{du_{kl}^{(i)}(t)}{dt} &= -\lambda_{kl} u_{kl}^{(i)} \\ &\quad - \frac{1}{L^2} \sum_{\alpha,\beta=0}^{L-1} [F(x_{\alpha\beta} + U_{\alpha\beta}(t)) - G(x_{\alpha\beta})] \frac{\partial F}{\partial x_i}(x_{\alpha\beta} + U_{\alpha\beta}(t)) \phi_{kl}(x_{\alpha\beta}). \end{aligned}$$

Writing

$$(5) \quad Z_{\alpha\beta}^{(i)}(t) = [F(x_{\alpha\beta} + U_{\alpha\beta}(t)) - G(x_{\alpha\beta})] \frac{\partial F}{\partial x_i}(x_{\alpha\beta} + U_{\alpha\beta}(t)),$$

the second term in (4) is simply $(\pi(Z^{(i)}(t)))_{kl}$. The choice of initial point zero is motivated by the fact that the mapping generated by the solution is expected to be in some neighborhood of the identity map. The symmetric difference approximation of the derivatives of F was used.

It was found that minimizing all $2(2N)^2$ coefficients at once for large N is not the best approach. It might be preferable to start out in a low-dimensional space and gradually to increase the dimension until the desired maximum dimension is reached. The initial point in each space is then taken as the local minimum obtained in the previous space. This procedure is also faster. Heuristically, this can be interpreted as matching global features in low-dimensional spaces and moving on to the finer details as the dimension increases.

The algorithm is described as follows.

- (i) Set $N = 0$, initial condition $u(0) = 0$.
- (ii) Find local minimum \hat{u}_N for J_N , starting from $u(0)$.
- (iii) Set $N \rightarrow N + 1$ and $u_{kl}^{(i)}(0) = \begin{cases} (\hat{u}_N^{(i)})_{kl} & \text{for } -N < k, l \leq N, \\ 0 & \text{otherwise.} \end{cases}$
- (iv) Go to (ii).

Step (ii) is carried out by a simple Euler method.

- (a) Fix time step dt , set $t = 0$.
- (b) Generate $U(t)$ from $(u^{(1)}(t), u^{(2)}(t))$ -inverse DFT, $U^{(i)} = \pi^{-1}(u^{(i)})$ or

$$U_{\alpha\beta}^{(i)}(t) = \sum_{k,l=-N+1}^N u_{kl}^{(i)}(t) \phi_{kl}(x_{\alpha\beta}), \quad \alpha, \beta = 0, \dots, L - 1.$$

- (c) Carry out quadrature (in (4)) for each eigenvector ϕ_{kl} -DFT, $V^{(i)}(t) = \pi(Z^{(i)}(t))$ or

$$V_{kl}^{(i)}(t) = \frac{1}{L^2} \sum_{\alpha,\beta=0}^{L-1} F(x_{\alpha\beta} + U_{\alpha\beta}(t)) \frac{\partial F}{\partial x_i}(x_{\alpha\beta} + U_{\alpha\beta}(t)) \phi_{kl}(x_{\alpha\beta}).$$

- (d) Add the linear term and carry out Euler step.

$$u_{kl}^{(i)}(t + 1) = u_{kl}^{(i)}(t) - dt \cdot [V_{kl}^{(i)}(t) + \lambda_{kl} u_{kl}^{(i)}(t)].$$

- If difference is smaller than tolerance, set $\hat{u}_N = u(t + 1)$ and go to (iii).
- (e) Go to (b).

The time-consuming parts of the algorithm are steps (b) and (c). One option is to apply a fast Fourier transform (FFT) and an inverse FFT. The dimensions of these FFTs are determined by the size of the pixel lattice on which F and G are defined, i.e., 128×128 or 256×256 . However, when N is very small, or when the dimensions of the pixel lattice are not powers of two, it may be faster to actually carry out the quadrature in step (c) for those frequencies that are being updated and to carry out the summation in step (b) for all points (α, β) on the lattice.

In some cases it is possible to carry out the quadrature in (c) on a coarser lattice than the original pixel lattice; then, of course, the summation in (b) is only done for the points on the coarse lattice. This is particularly true when F and G are smooth functions. This is related to the multigrid approach suggested in [11]. However, in [11] the number of unknowns is

equal to the number of points used and this seems to be insufficient. This is also the case for the pseudospectral methods described below. In the experiments described here the algorithm did not produce much improvement for $N > 10$, and at low frequencies the coarse lattice quadrature method is indeed faster than the FFT method.

In some experiments it was useful to actually smooth out the template and the data through some low-pass filter (see Figs. 3 and 4). This tends to single out global topological features and eliminate local ones. In this case the coarse lattice quadrature is very appropriate. For images 128×128 , one could work with a 16×16 lattice for the first four or five frequencies and get good results.

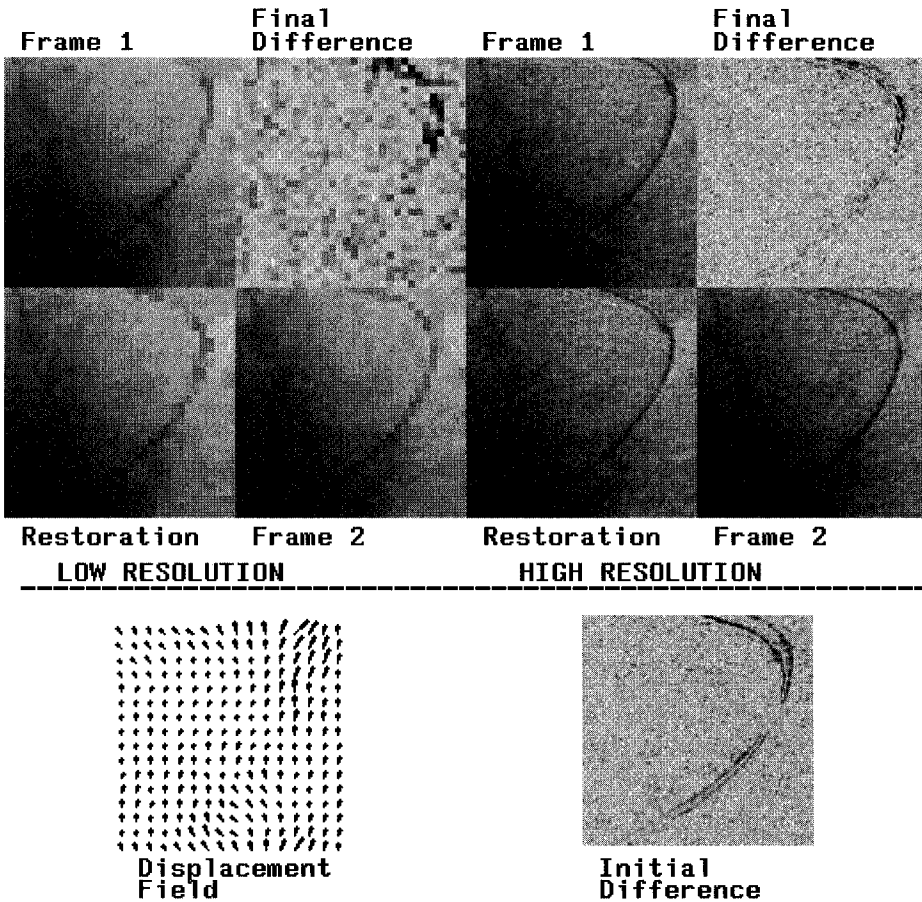


FIG. 3. Frames 1 and 2 are two subsequent pictures from an x-ray of a catheter inserted into an artery. Frame 1 is used as the template and Frame 2 as the data. The displacement field is found by using the wavelet method at the low resolution, and then applied to the high-resolution template (Frame 1). The black line in the restoration is almost perfectly aligned with the black line in the second frame.

If the full lattice has $L \times L$ pixels and the coarser lattice has $K \times K$ pixels, the number of floating-point multiplications used in this method is of order $K^2 \cdot N^2$, with N being the number of frequencies, whereas using the full FFT would be approximately $L^2 \cdot \log L^2$. Thus, depending on the number of frequencies to be used, it is possible to choose which is the most appropriate. As a rule, K , the number of quadrature points, should be greater than N , the

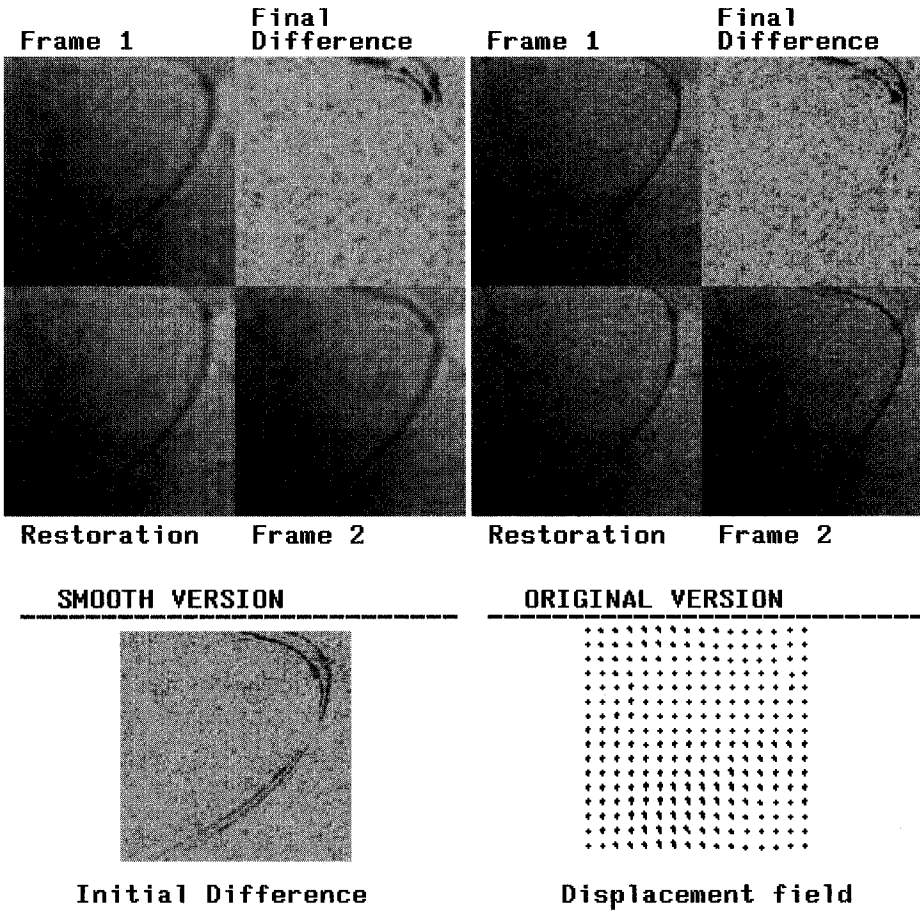


FIG. 4. The same setting as in Fig. 3. The restoration is done using the Fourier basis. The line in the restoration is not aligned at the upper right-hand corner.

number of frequencies being solved. For high frequencies the coarse lattice approximation is therefore not suitable, in which case it is clearly advantageous to use the FFT.

Another method is the pseudo-spectral method described by Gottlieb, Hussaini, and Orszag in [4]. Since one does not expect to solve, say, for $N > K$ (namely, the solution that is assumed to be in $H_K \times H_K$) using the periodic version of the sampling theorem, one can write

$$U^{(i)}(x) = \sum_{\alpha, \beta=0}^{K-1} U_{\alpha\beta}^{(i)} S_{\alpha\beta}(x),$$

where $S_{\alpha\beta}(x)$ are trigonometric polynomials and $x_{\alpha\beta}, \alpha, \beta = 0, \dots, K - 1$ are equally spaced points on the unit square called the collocation points. Thus

$$\Delta^2 U^{(i)}(x) = \sum_{\alpha, \beta=0}^{K-1} U_{\alpha\beta}^{(i)} \Delta^2 S_{\alpha\beta}(x).$$

If the quadrature in (3) is carried out on the $K \times K$ lattice of collocation points, then the

gradient descent equation has the form

$$\begin{aligned} \frac{dU^{(i)\gamma\delta}(t)}{dt} = & - \sum_{\alpha\beta} U_{\alpha\beta}^{(i)}(t)(\Delta^2 + \varepsilon)S_{\alpha\beta}(x_{\gamma\delta}) \\ & - [F(x_{\gamma\delta} + U_{\gamma\delta}(t)) - G(x_{\gamma\delta})] \frac{\partial F}{\partial x_i}(x_{\gamma\delta} + U_{\gamma\delta}(t)), \end{aligned}$$

for $\gamma, \delta = 0, \dots, K - 1$. Now since the minimization is taking place in the space domain, the coupling between the equations occurs in the smoothing term and not in the data term. The derivatives of the functions S at all collocation points can be calculated off-line and stored as a matrix.

The discrete time iterations for this ODE require only one time-consuming step, the summation, as opposed to the previous schemes that had two time-consuming steps. If a $K \times K$ lattice of collocation points is used, the number of floating-point multiplications is approximately K^4 .

The disadvantage of this method lies in the fact that the number of frequencies being updated is equivalent to the number of collocation points used. Thus if only very low frequencies are to be updated, very few collocation points are used and they are insufficient to obtain a good result. On the other hand, if more collocation points are used, the solution obtained involves higher frequencies at the start and this might be undesirable. There is also no natural way to gradually increase the dimension of approximation.

As mentioned in the Introduction, a further possibility would be to linearize the functional by substituting $F(x) + \nabla F(x) \cdot U(x)$ for $F(x + U(x))$ in $I(U)$. Then we obtain a quadratic form in $U(x)$, or in the coefficients u_{kl} , which has a unique minimum. The Euler equations for this functional are precisely those suggested by Horn and Schnuck [5]:

$$\begin{aligned} \alpha(\Delta + \varepsilon)^2 U^{(1)} + (\nabla F \cdot U) \frac{\partial F}{\partial x_1} &= (G(x) - F(x)) \frac{\partial F}{\partial x_1}, \\ \alpha(\Delta + \varepsilon)^2 U^{(2)} + (\nabla F \cdot U) \frac{\partial F}{\partial x_2} &= (G(x) - F(x)) \frac{\partial F}{\partial x_2}. \end{aligned}$$

Experiments with this option have not led to satisfactory results; see §4 and Fig. 2.

The disadvantage of the various spectral methods described above is the use of the Fourier basis whose functions have global support. This makes local changes in U or in the mapping ϕ difficult to achieve. A very attractive alternative is the wavelet basis.

3. The wavelet method. Using a periodic wavelet basis obtained from certain types of compactly supported wavelets, it is possible to rewrite the functional J in such a way that the same regularization is achieved with a different bilinear form. First, a brief survey of various results regarding these wavelets. For a detailed description of the discrete wavelet transform used in the experiments see the Appendix.

Periodic wavelets. Let ψ be a compactly supported wavelet as constructed by Daubechies in [3] with support in $[-R, R]$. Let ϕ be the corresponding function that generates the multiresolution analysis. Observe that ϕ has compact support on $[0, 2R]$. Define $\psi_{nk}^P(x) = 2^{n/2} \sum_{l=-\infty}^{\infty} \psi(2^n(x-l) - k)$, and $\phi_{nk}^P(x) = 2^{n/2} \sum_{l=-\infty}^{\infty} \phi(2^n(x-l) - k)$ for $0 \leq x \leq 1$. All the functions with superscript P are periodic with period 1, and if $k = 2^n$, then $\psi_{nk}^P = \psi_{n0}^P$, similarly for ϕ^P . The number of terms in these sums is determined by the support of ψ and is bounded by $2R$. For $n > r = \log_2 R + 1$ there are at most two terms in the sum.

For small n the functions ψ_{nk}^P cannot be expressed as scales and shifts of $\psi_{00}^P(x) \equiv \psi^P$. However, for $n > r$, we have $\psi_{(n+1)k}^P(x) = \sqrt{2}\psi_{n0}^P(2x - k)$, where the argument is considered modulo 1 or on the unit circle. Moreover, the family of functions ψ_{nk}^P for $n = 0, 1, \dots$ and

$k = 0, 1, \dots, 2^n - 1$ together with the function $\phi_{00}^P \equiv \mathbf{1}$ form an orthonormal basis. Let $V_0 = \text{span}\{\phi_{00}^P\}$ (i.e., the constant functions) and $W_n = \text{span}\{\psi_{nk}^P\}_{k=0}^{2^n-1}$. Using the fact that the ordinary wavelet basis spans $L_2(R)$, it is not hard to verify that

$$L_2(T^1) = V_0 \oplus \bigoplus_{n=1}^{\infty} W_n,$$

where T^1 is the one-dimensional torus. All that is necessary is to extend a periodic function f to a function $F \in L^2(R)$ by making $2R$ copies to the right and to the left of the unit interval and leaving the rest zero. Expanding F in the ordinary wavelet basis, we find that the part supported on $[0, 1]$ can be given in terms of the functions $\mathbf{1}$ and ψ_{nk}^P ; see [9, Chap. III, §11].

By [3] for sufficiently large R , the functions ψ and ϕ are twice differentiable and the first two moments of ψ are zero. Hence by [9, Chap. II, Thm. 8], $f \in H_2(T)$ if and only if

$$C(f, f) = f_{00}^2 + \sum_{n=1}^{\infty} \sum_{k=0}^{2^n-1} f_{nk}^2 (1 + 4^{2n}) < \infty,$$

where $f_{00} = \int f(x)dx$ and $f_{nk} = \int f(x)\psi_{nk}^P(x)dx$. This implies, via the closed graph theorem, that the Hilbert norm $C(f, f)$ defined above is equivalent to the Sobolev norm.

In two dimensions let

$$\begin{aligned} \Phi_{nkl}^P(x) &= \phi_{nk}^P(x_1)\phi_{nl}^P(x_2), & \Psi_{nkl}^{P,a}(x) &= \phi_{nk}^P(x_1)\psi_{nl}^P(x_2), \\ \Psi_{nkl}^{P,b}(x) &= \psi_{nk}^P(x_1)\phi_{nl}^P(x_2), & \Psi_{nkl}^{P,c}(x) &= \psi_{nk}^P(x_1)\psi_{nl}^P(x_2). \end{aligned}$$

Let V_0 denote the constant functions and

$$W_n^{(a)} = \text{span} \left\{ \Psi_{nkl}^{P,a} \right\}_{k,l=0}^{2^n-1}, \quad n = 1, \dots, \infty,$$

similarly for b and c . Setting $W_n = W_n^{P,a} \oplus W_n^{P,b} \oplus W_n^{P,c}$, we can show that

$$L_2(T^2) = V_0 \oplus \bigoplus_{n=1}^{\infty} W_n.$$

As in one dimension, $f \in H^2(T^2)$ if and only if

$$C(f, f) = f_{00}^2 + \sum_{n=1}^{\infty} \sum_{k,l=0}^{2^n-1} (1 + 4^{2n}) \left((f_{nkl}^{(a)})^2 + (f_{nkl}^{(b)})^2 + (f_{nkl}^{(c)})^2 \right) < \infty,$$

where $f_{nkl}^{(a)} = \int f(x)\psi_{nkl}^{P,a}(x)dx$, similarly for b and c . Consequently, it is possible to redefine $\varepsilon(U, U)$ in the functional J as $C(U^{(1)}, U^{(1)}) + C(U^{(2)}, U^{(2)})$ to obtain the same regularization as before. Now the finite dimensional spaces are given by

$$H_N = V_0 \oplus \bigoplus_{n=0}^{N-1} W_n.$$

Thus the regularization on the infinite dimensional space is the same as in the Fourier method; however, the sequence of approximations is different, and hence should lead to different types of solutions.

Observe that H_N is of dimension $2^N \times 2^N$. Moreover, for $f \in H_N$, the coefficients can be arranged in a $2^N \times 2^N$ array C so that $C_{00} = f_0$ and

$$C_{kl} = \begin{cases} f_{nkl}^{(a)} & \text{for } 2^n \leq k < 2^{n+1} & \text{and } 0 \leq l < 2^n, \\ f_{nkl}^{(b)} & \text{for } 0 \leq k < 2^n & \text{and } 2^n \leq l < 2^{n+1}, \\ f_{nkl}^{(c)} & \text{for } 2^n \leq k < 2^{n+1} & \text{and } 2^n \leq l < 2^{n+1}, \end{cases}$$

for $n = 1, \dots, N - 1$.

Practical considerations. The algorithm itself proceeds very much like the one described in the previous section except that the Fourier transform is replaced by the two-dimensional wavelet transform. In practice, we are working with discrete data so that we use the discrete two-dimensional wavelet transform developed by Mallat in [8].

This transform takes the data D given on a discrete $2^N \times 2^N$ lattice as representing the coefficients of some function $f \in H_N$ on the unit square with respect to the functions Φ_{Nkl}^P , $k, l = 0, \dots, 2^N - 1$. Think of f as an interpolation between the data values D_{ij} given on the discrete lattice using the compactly supported functions Φ_{nkl}^P . As above, the function f can be decomposed into its coefficients with respect to the wavelet basis. Let C denote this new $2^N \times 2^N$ array of coefficients, and write $C = W(D)$. Mallat [8] provides a very simple and fast algorithm for calculating $W(D)$, the forward wavelet transform, and $W^{-1}(C)$, the inverse wavelet transform. For the sake of completeness this is described in detail in the Appendix.

Now the algorithm described previously is modified as follows. Let $u^{(i)}$ describe the wavelet coefficients of $U^{(i)}$, i.e., $u^{(i)} = W(U^{(i)})$. In step (iii) we now have

$$u_{kl}^{(i)}(0) = \begin{cases} (\hat{u}_N)_{kl} & \text{for } 0 \leq k, l < 2^N - 1, \\ 0 & \text{for } 2^N \leq k < 2^{N+1} \text{ or } 2^N \leq l < 2^{N+1}. \end{cases}$$

In other words, when increasing the level of approximation $3 \cdot 2^N$ zeros are added for each component of the displacement field. In step (b) we have $U^{(i)}(t) = W^{-1}(u^{(i)}(t))$ and in step (c) $V^{(i)}(t) = W(Z^{(i)}(t))$ with $Z^{(i)}(t)$ as in (5). In step (d) the eigenvalues λ_{kl} are replaced by $\eta_{kl} = 1 + 4^{2n}$ for $2^n \leq k < 2^{n+1}$, $0 \leq l < 2^n$, or $0 \leq k < 2^n$, $2^n \leq l < 2^{n+1}$, or $2^n \leq k < 2^{n+1}$, $2^n \leq l < 2^{n+1}$.

In this approach the smoothing term is, of course, decoupled as before. In addition, since at higher frequencies ψ_{nk} have small support, the data parts of the equations are only *locally coupled*. This is in contrast to the Fourier approach in which *all* the equations are coupled through the data part.

This points to a clear advantage of the wavelet approach. It allows for local updates in problematic regions where the difference between $F(x + U(x))$ and $G(x)$ is still large, which has a relatively small effect on other regions where the difference is small.

The periodic setting we have chosen greatly facilitates the discrete algorithm by eliminating the need to store extra boundary terms at each level. Thus the size of the transformed data at each level of resolution is precisely that of the original data.

In practice we have used $R = 3$, which does not generate a wavelet smooth enough to apply the above-mentioned theorems; however, it performed well enough for our purposes.

The wavelet algorithm was tried using only the discrete transform. Due to its speed and ease of implementation there was no need to try the quadrature method for low-frequency coefficients. Moreover, instead of smoothing the data G and the template F as in the Fourier method, we take a lower resolution version of both, which consists of smoothing and sampling at the coarser grid (see the Appendix). The field U is found on the coarser grid. This field is then interpolated to the fine grid through the inversion formula (6) in the Appendix with the y coefficients set to zero (see Fig. 3). Again, this is similar to multigrid techniques; however, we did not apply it to a cascade of grids or continue the algorithm at the fine grid. This was merely a way to help the algorithm to “see” similar topological structures.

4. Description of experiments. In the first experiment (Fig. 1), the transformation between one x-ray of a hand, the template, to another, the data, is found and is used to automatically identify the location of important landmarks of the hand in the data image. The white lines connect points in the template to the points they are mapped to by the displacement field.

In this case, the tip of the index finger was successfully located in the data, whereas the tip of the little finger was slightly misplaced. The restoration has definitely succeeded in correcting the widths of the fingers and the palm. Observe that the fingers in the template are wider than those in the data, as is the palm of the hand. The absolute value of the initial difference between the images is shown as compared to the absolute value of the difference between the restoration image and the data. The black area indicates large differences.

In the second experiment (Fig. 5), the restoration on the right represents the result using a fixed dimension, whereas the restoration on the left used a sequence of increasing dimensions. The maximum dimension used was $N = 7$. Using the increasing dimension method not only produces a better result, but is also much faster when the Fourier transform is calculated via quadrature only for the frequencies being updated.

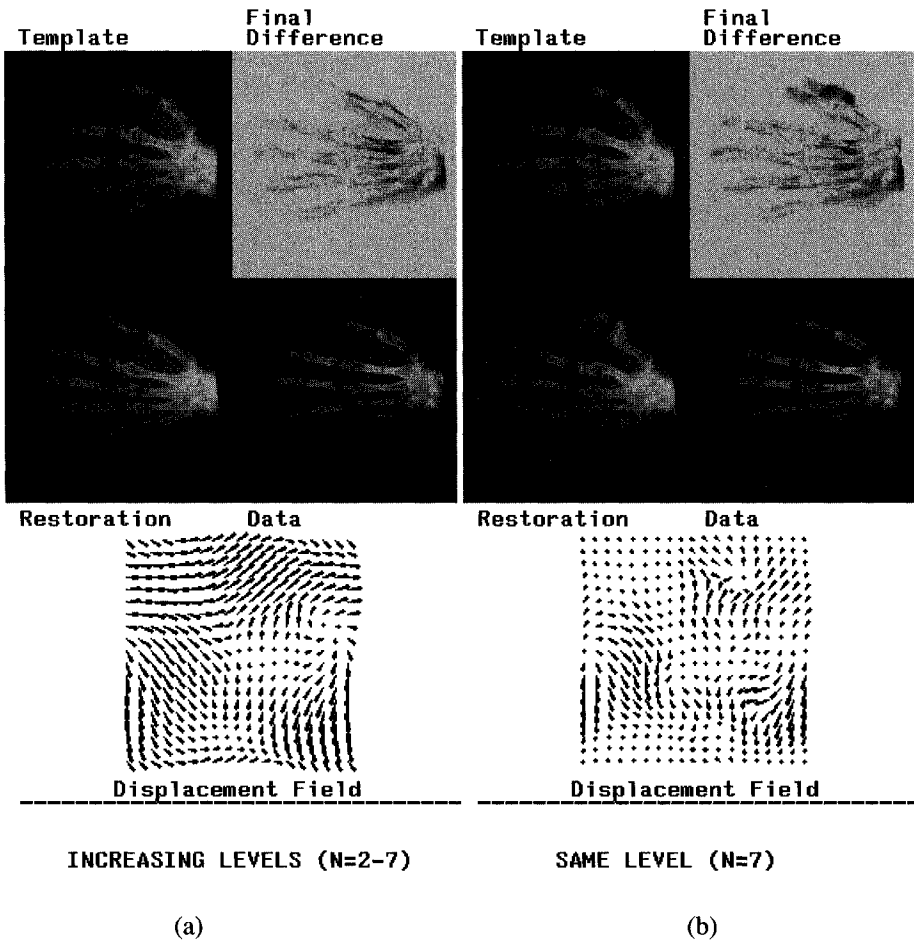


FIG. 5. The restoration was done using the Fourier method. Using (a) increasing levels of approximation starting at $N = 2$ up to $N = 7$, and (b) using a fixed level $N = 7$.

In the third experiment, we use a synthetic image consisting of two “humps” for the template F . The data G was created by composing F with some field U . The field U was generated by drawing the coefficients of the Fourier basis from independent Gaussian random variables with variances $1/\lambda_{nm}$, for $-9 \leq n, m \leq 10$. On the left in Fig. 2, we have the restoration process using the Fourier basis and on the right using the wavelet basis. Observe

that the final field obtained by both methods is very similar. Again, it is possible to see that the topological landmarks — the maxima, the saddle points, etc. — are mapped into each other by the field U . At the bottom we have the restoration using the linearized equations. These did not do nearly as well in spite of the fact that the mean square error in the equation was down to 10^{-5} .

The fourth experiment (Fig. 6) shows what happens when we attempt to use a template with the wrong topological features (Template 2) as compared to the correct template (Template 1). The data in this case was the same as in the previous two pictures with added independently and identically distributed Gaussian noise of variance 0.2, when the grey level values are scaled to the interval $[0, 1]$. Even though the restoration using the wrong template is fairly good, the displacement field is highly irregular when compared to the displacement field obtained by using the correct template. This might indicate a method of determining which of several possible templates corresponds to the objects in the noisy image.

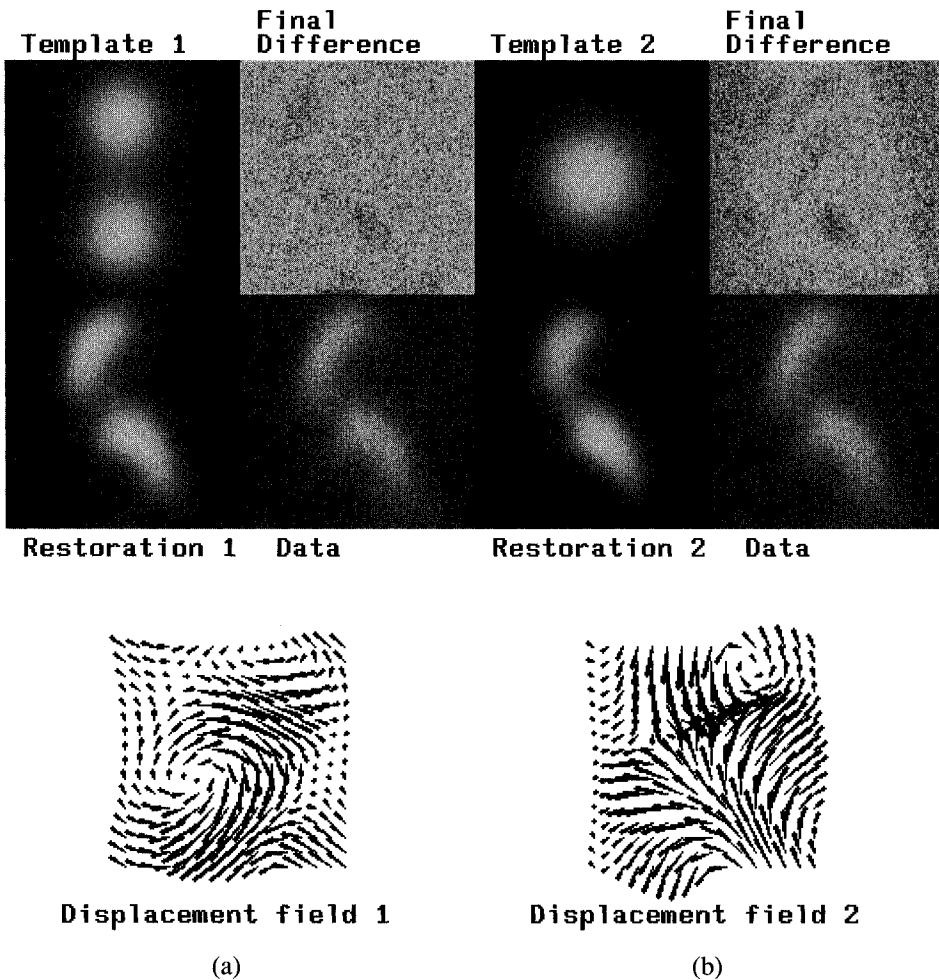


FIG. 6. Gaussian independently and identically distributed noise was added to the data. In (a) the correct template was used, and in (b) a different template was used (one hump instead of two).

The fifth experiment illustrates the possible advantages of the wavelet basis. The template F is a subimage from a sequence of x-rays of a catheter that has been inserted into a coronary

artery. The data G is the corresponding subimage of the x-ray that follows in the sequence. The idea is to use the algorithm to find the movement of the catheter between the two images. In this experiment the template F is larger than what is actually seen in the image. In other words, the function F is defined outside the unit square. When $x + U(x)$ lies outside the unit square, the value of F is taken from that point and not from the corresponding point on the torus. Thus we did not wrap around when composing the template with the mapping. This allows the algorithm to find possible movement *in and out* of the frame under consideration, and is, of course, much better adapted to the issue of movement compensation between consecutive movie frames.

Figure 3 shows the result using the wavelet basis. On the left side are the low resolution versions of the template, Frame 1, and of the data, Frame 2, together with the restoration obtained by using these lower resolution versions in the algorithm. On the right side are the original Frame 1 and Frame 2 together with the restoration obtained by using the displacement field from the low-resolution calculation. Most of the movement between these two consecutive frames has occurred in the upper right-hand corner and, indeed, the restoration has found that movement. Observe that in the low-resolution versions the black lines are widened and thus begin to overlap. This is what generates a gradient for the algorithm to proceed along.

Figure 4 shows the result of using the Fourier basis. The smooth version is on the right and the high resolution is on the left. Observe that this time the change in the upper right-hand corner was not found by the restoration procedure. Apparently, the wavelet basis performed better due to its ability to create a local change in the field without affecting other parts of the image.

In this experiment both the data *and* the template images are quite noisy. In the Fourier method the Fourier transforms of the template and the data are low-pass filtered and retransformed to the original space domain. They are, therefore, much smoother than in the wavelet case where a low-resolution version of the two is used, i.e., a low-pass filter that is not retransformed to the original resolution. This explains why the final low-resolution difference image in Fig. 3 is much noisier than the corresponding smooth version difference image in Fig. 4.

It should be noted that this movement analysis of the catheter is done without any pre-processing of the images to find the line or edge corresponding to the catheter. The algorithm was "helped" only by having it run on the low-resolution transform of the images. In such a way the "canyons" corresponding to the lines become wide valleys that overlap and enable the algorithm to draw them into one another. In other words, if the high resolution images are used, small changes of the field do not produce a better mean square error because the error along both canyons remains the same. Thus there is no clear gradient in any direction. On the other hand when the "valleys" overlap, the direction in which the field could reduce the error is clear.

In the sixth experiment (Fig. 7), we tried to find the movement between two subsequent frames of x-rays of the coronary arteries themselves. The direction of movement varies between different parts of the frame and the wavelet method managed to accommodate these local changes. In this experiment the actual difference between the two frames is compared to the actual difference between the restoration and Frame 2. The grey levels indicate values close to zero. Black or white values indicate large differences.

5. Conclusion. We have presented a nonlinear functional whose minimizers represent the mapping that transforms one image or function into another. The minimizers make sense only in so far as the two functions considered as surfaces have similar topographies, so that one function may be considered as a template for the other.

The functional is regularized using two different choices of a bilinear form. The first form is generated by a differential operator and has a spectral representation using the Fourier basis.

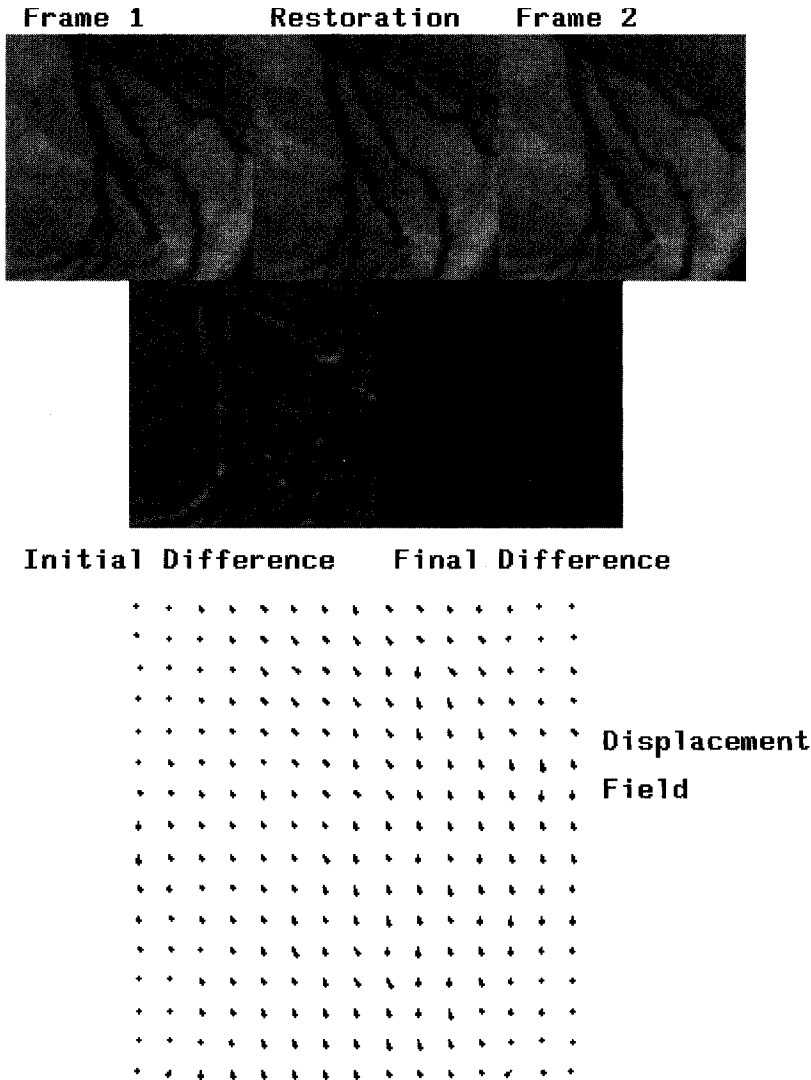


FIG. 7. Frames 1 and 2 are two subsequent pictures from an x-ray movie of the coronary arteries. The displacement field corresponds to the movement between the two frames. The initial difference and the final difference pictures represent the actual difference. Grey values are close to zero and white and black values correspond to large differences. The restoration was done using the wavelet method.

The second form is directly defined in the spectral domain using a wavelet basis of compact support. Both bilinear forms generate the same type of regularization, namely, they constrain the solutions to the same Sobolev space.

The variational problem is solved in terms of the expansion coefficients of the unknown map in terms of the chosen basis. The solution starts at low frequency and gradually moves up. The experimental results clearly indicate that the wavelet basis is more flexible and permits local changes in the mapping.

Appendix. For the sake of completeness we briefly describe the discrete wavelet transform so that the interested reader can code it without further reading. We start with the one-dimensional case.

Let $N = 2^n$ and let $x^{(0)} = (x_0, \dots, x_{N-1})$ be the data. Given two vectors $g = (g_0, \dots, g_{2R-1})$, and $h = (h_0, \dots, h_{2R-1})$, let $\tilde{x} = h * x^{(0)}$ and $\tilde{y} = g * x^{(0)}$, i.e.,

$$\tilde{x}_k = \sum_{j=0}^{2R-1} h_j x_{[(k+j) \bmod N]}, \quad \tilde{y}_k = \sum_{j=0}^{2R-1} g_j x_{[(k+j) \bmod N]},$$

for $k = 0, \dots, N-1$. Now set $x_i^{(1)} = \tilde{x}_{2i}$ and $y_i^{(1)} = \tilde{y}_{2i}$ for $i = 0, \dots, N/2-1$. Alternatively, we write $x^{(1)} = H_n x^{(0)}$ and $y^{(1)} = G_n x^{(0)}$, where H_n, G_n are linear mappings from R_N to $R^{N/2}$. As long as $N > 2R$, it is possible to write the $N/2 \times N$ matrix for G_n as follows:

$$G_n = \begin{pmatrix} g_0 & g_1 & g_2 & g_3 & \dots & g_{2R-3} & g_{2R-2} & g_{2R-1} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & g_0 & g_1 & \dots & \dots & \dots & g_{2R-3} & g_{2R-2} & g_{2R-1} & \dots & 0 & 0 \\ & & & \vdots & & & & & & & & & \\ g_2 & g_3 & \dots & \dots & \dots & g_{2R-1} & 0 & 0 & 0 & 0 & \dots & g_0 & g_1 \end{pmatrix}.$$

The $N/2 \times N$ matrix for H_n is given in the same way using the vector h . (Observe that we have indexed the matrices according to the log of the dimension.) This is the first level of the transform. Using the properties of multiresolution analysis and other techniques (see [3] and [8]) it is possible to find vectors g and h with the following properties.

- (i) $\sum_{i=0}^{2R-1} g_i = 0$ and $\sum_{i=0}^{2R-1} h_i = \sqrt{2}$,
- (ii) $g_i = (-1)^n h_{2R-1-i}$,
- (iii) $H_n^t H_n + G_n^t G_n = I$,
- (iv) $H_n G_n^t = 0$.

The first item implies that the H_n matrix is a smoothing operator and that the G_n matrix is a difference operator. The third item is the inversion formula. The fourth item is an orthogonality condition. Given two vectors x, y of length $N/2$, then the corresponding vectors in the full resolution space are orthogonal, i.e., $\langle H_n^t x, G_n^t y \rangle = 0$.

The second level of the transform is given by $x_i^{(2)} = h * x_{2i}^{(1)}$ and $y_i^{(2)} = g * x_{2i}^{(1)}$, for $i = 0, \dots, N/4 - 1$. $y^{(1)}$ remains unchanged. Thus, at each level, the smoothed version $x^{(i)}$, which is of length 2^{n-i} , is decomposed into yet a smoother version $x^{(i+1)}$ and the difference component $y^{(i+1)}$ both of length 2^{n-i-1} . Clearly, this procedure comes to an end at step $i = n$, where $x^{(n)}$ and $y^{(n)}$ are scalars. The sequence of vectors $x^{(1)}, \dots, x^{(n)}$ are called *lower resolution versions* of the original vector $x^{(0)}$.

From the inversion formula it follows that

$$x^{(i)} = H_{n-i}^t H_{n-i-1}^t \dots H_1^t x^{(n)} + \sum_{l=i}^{n-1} H_{n-i}^t \dots H_{n-l+1}^t G_{n-l}^t y^{(l)}.$$

In Daubechies [3], numbers for the vectors h and g are calculated so as to satisfy the four conditions listed above (see [3, p. 980]). Moreover, Daubechies shows that the wavelet corresponding to these numbers satisfies certain smoothness conditions. In terms of the inversion formula above this means that if we set $x^{(n)} = 1$ and $y^{(j)} = 0$ for $j = 1, \dots, n$, then the vectors

$$x^{(i)} = H_{n-i}^t H_{n-i-1}^t \dots H_1^t \mathbf{1}$$

for small values of i are smooth. It should be noted that since we are working here in the periodic setting, $x^{(0)}$ would be constant.

In two dimensions the wavelet transform is constructed using the one-dimensional operators described above. An initial $N \times N$ matrix $x^{(0)}$ is decomposed into four components. First, the one-dimensional transform is carried out on the columns, $A = H_n x^{(0)}$ and $B = G_n x^{(0)}$, where A, B are $N/2 \times N$ matrices. Then the one-dimensional transform is carried out on the rows of A and B or on the columns of A^t and B^t , and we obtain

$$\begin{aligned} x^{(1)} &= H_n A^t = H_n (H_n x^{(0)})^t, & y_a^{(1)} &= G_n A^t = G_n (H_n x^{(0)})^t, \\ y_b^{(1)} &= H_n B^t = H_n (G_n x^{(0)})^t, & y_c^{(1)} &= G_n B^t = G_n (G_n x^{(0)})^t. \end{aligned}$$

The low-resolution version of $x^{(0)}$ is $x^{(1)}$. The transform continues by operating on $x^{(1)}$ with the matrices G_{n-1}, H_{n-1} . Finally, the inversion formula has the form

$$(6) \quad x^{(0)} = G_n^t (G_n^t y_c^{(1)} + H_n^t y_b^{(1)})^t + H_n^t (G_n^t y_a^{(1)} + H_n^t x^{(1)})^t.$$

Acknowledgments. I would like to express my gratitude to David Gottlieb for his suggestions and encouragement.

REFERENCES

- [1] Y. AMIT, U. GRENANDER, AND M. PICCIONI, *Structural image restoration through deformable templates*, J. Amer. Statist. Assoc., 86 (1991), pp. 376–387.
- [2] R. BAJCY AND S. KOVACIC, *Multiresolution Elastic Matching*, Computer Vision, Graphics, and Image Processing, 46 (1989), pp. 1–21.
- [3] I. DAUBECHIES, *Orthonormal bases of compactly supported wavelets*, Comm. Pure Appl. Math., XLI (1988), pp. 909–996.
- [4] D. GOTTLIEB, M. Y. HUSSAINI, AND A. O. ORSZAG, *Theory and applications of spectral methods*, in Spectral Methods for Partial Differential Equations, R. G. Voigt, D. Gottlieb, and M. Y. Hussaini, eds., Society for Industrial and Applied Mathematics, Philadelphia, PA, 1984.
- [5] B. K. P. HORN AND B. G. SCHNUCK, *Determining optical flow*, Artificial Intelligence, 17 (1981), pp. 185–203.
- [6] T. S. HUANG AND R. Y. TSAI, *Image sequence analysis: Motion estimation*, in Image Sequence Analysis, T. S. Huang, ed., Springer-Verlag, New York, 1981.
- [7] Z. JIN AND P. MOWFORTH, *A discrete approach to signal mapping*, Tech. Report TIRM-88-036, The Turing Institute, Glasgow, Scotland, 1988.
- [8] S. MALLAT, *A theory for multiresolution signal decomposition: the wavelet representation*, in IEEE Trans. on Pattern Analysis and Machine Intelligence, 11 (1989), pp. 674–693.
- [9] Y. MEYER, *Ondelettes et Operateurs*, Hermann, Paris, 1990.
- [10] H. H. NAGEL, *Displacement vectors derived from second-order intensity variations in image sequences*, Computer Vision, Graphics, and Image Processing, 21 (1983), pp. 85–117.
- [11] D. TERZOPOULOS, *Image analysis using multigrid relaxation methods*, in IEEE Trans. on Pattern Analysis and Machine Intelligence, 8 (1986), pp. 129–139.