11.6.3 *Steepest descent interpretation*

In this section we give another straightforward interpretation of Thirion's approach. The overall goal is to minimize the distance measure $\mathcal{D}[R, T; u]$; cf., eqn (8.2). Starting with a smooth displacement u one might use a steepest descent method $\partial_t u = -f(\cdot, u)$, see Theorem 8.1, or in a time discrete setting,

$$u^{(k+1)} = u^{(k)} + \tau f(\cdot, u^{(k)}).$$

The problem is that f is in general not smooth, such that the update and hence $u^{(k+1)}$ might be non-smooth. The trick is to project the update onto a smooth space, e.g., by convolving with a Gauss kernel. This gives

$$u^{(k+1)} = K_\sigma * (u^{(k)} + \tau f(\cdot, u^{(k)})).$$

11.7 Discussion of diffusion registration

The main drawback of diffusion registration is that although it is obvious to measure smoothness by oscillations of the gradients, it is not physical. Even though each component u_ℓ of the displacement can be viewed as a solution of a particular heat equation, a physical interpretation for the vector field u is missing. In our experience, however, the non-physical behavior of the method can hardly be detected in "real-life" applications.

In the registration process, the spatial directions are coupled only through the forces. It is this property that can be viewed as one major advantage of diffusion registration. The spatial decoupling allows for a block diagonalization. In addition, for each block the AOS scheme (cf., Section 11.3) presents a fast and stable solution technique of linear complexity. This makes diffusion registration a very attractive registration scheme, in particular for high-dimensional image data.

12

CURVATURE REGISTRATION

Elastic, fluid, and diffusion registration are sensitive with respect to affine linear displacements; see, e.g., Section 10.6. In particular, for dimension $d = 2$ and

$$u(x) = Cx + b, \quad \text{with} \quad C \in \mathbb{R}^{2\times 2}, \quad b \in \mathbb{R}^2,$$

for the elastic potential of u, we have

$$\mathcal{P}[u] = \mathcal{S}^{\text{elas}}[u] = \int_\Omega \mu(c_{1,1}^2 + (c_{1,2} + c_{2,1})^2/2 + c_{2,2}^2) + \lambda(c_{1,1} + c_{2,2})^2/2\, dx.$$

Hence, for $\mu \neq 0$,

$$\mathcal{P}[u] = 0 \iff c_{1,1} = c_{2,2} = 0 \wedge c_{1,2} = -c_{2,1}.$$

Note that in the derivation of elastic registration (see Section 9.1), we explicitly decomposed the transformation into rigid and non-rigid parts. Thus, from this modeling, the rigid parts need to be pre-registered.

For diffusion registration, we have

$$\mathcal{S}^{\text{diff}}[u] = \int_\Omega c_{1,1}^2 + c_{1,2}^2 + c_{2,1}^2 + c_{2,1}^2 dx$$

and $\mathcal{S}^{\text{diff}}[u] = 0 \iff C = 0$. Since diffusion registration penalizes the norm of the gradient, this property is a direct consequence of the regularizing term.

As a consequence, for all these non-linear registration techniques an affine linear pre-registration is unavoidable. In order to circumvent this additional pre-registration we introduce a novel regularizing term based on second order derivatives. Since the regularizer is related to *curvature*, the novel registration is called *curvature registration*; cf., Fischer & Modersitzki (2003).

The main point is not that the additional pre-registration becomes redundant but that the registration becomes less dependent on the initial position of the reference and template images.

Curvature registration is based on the distance measure $\mathcal{D}$ (cf., eqn (8.10)) and the regularizer

$$\mathcal{S}^{\text{curv}}[u] := \tfrac{1}{2} a[u, u], \tag{12.1}$$

where the bi-linear form a is defined by

$$a[u, v] = \sum_{\ell=1}^{d} \int_\Omega \Delta u_\ell \Delta v_\ell \, dx$$

and Neumann boundary conditions $\nabla u_\ell = \nabla\Delta u_\ell = 0$ for $x \in \partial\Omega$, $\ell = 1, \dots, d$, are imposed.

The integrand $(\Delta u_\ell)^2$ of $\mathcal{S}^{\text{curv}}$ might be viewed as an approximation to the curvature. Thus, the idea of the regularizer is to minimize the curvature of the components of the displacement.

12.1 Continuous and discrete bi-harmonic equations

The Euler–Lagrange equations for the joint functional of Problem 8.1 are summarized in Theorem 12.1.

Theorem 12.1 *The* Euler–Lagrange *equations for* $\mathcal{J}^{\text{curv}} = \mathcal{D}^{\text{SSD}} + \alpha\mathcal{S}^{\text{curv}}$, *where* $\mathcal{D}^{\text{SSD}}$ *is defined by eqn (8.10) and* $\mathcal{S}^{\text{SSD}}$ *is defined by eqn (12.1) are*

$$f(x, u(x)) + \alpha\Delta^2 u(x) = 0, \quad x \in \Omega, \tag{12.2}$$

$\nabla u_\ell = \nabla\Delta u_\ell = 0$ *for* $x \in \partial\Omega$, $\ell = 1, \dots, d$.

Proof Follows from Theorem 8.1 and, with $\widetilde{a}[\xi, \eta] = \int_\Omega \Delta\xi\Delta\eta\, dx$, from

$$\begin{aligned}
\widetilde{a}[\xi, \eta] &= \int_\Omega \Delta\xi\Delta\eta\, dx \\
&= \int_{\partial\Omega} \Delta\xi \langle\nabla\eta, n\rangle_{\mathbb{R}^d}\, dx - \int_\Omega \langle\nabla\Delta\xi, \nabla\eta\rangle_{\mathbb{R}^d}\, dx \\
&= -\int_{\partial\Omega} \eta \langle\nabla\Delta\xi, n\rangle_{\mathbb{R}^d}\, dx + \int_\Omega \eta\Delta^2\xi\, dx \\
&= \int_\Omega \eta\Delta^2\xi\, dx.
\end{aligned}$$

Here, Green's formula and the Neumann boundary conditions have been utilized. □

As an important fact, we note that the partial differential operator related to the smoother $\mathcal{S}^{\text{curv}}$ is just the second power of the partial differential operator arising in diffusion registration, cf., Theorem 11.1. As a consequence, we employ a numerical scheme for the discrete Euler–Lagrange equations

$$f(\vec{X}, \vec{U}) + \alpha I_d \otimes A^{\text{curv},d}\vec{U} = 0, \tag{12.3}$$

where $A^{\text{curv},d} = (A^{\text{diff},d})^\top A^{\text{diff},d} = (A^{\text{diff},d})^2$; see also eqn (11.4).

The Euler–Lagrange equations for the curvature registration functional is also known as the *bi-harmonic* equation; see, e.g., Hackbusch (1987, §5.3).

As a by-product of the analysis presented in Section 11.2 we have the following corollary; see also Theorem 11.3.

Corollary 12.2 *Let* S *be a* d*-dimensional, symmetric matrix stencil and* $A^{(d)}$ *be the matrix representation of the convolution with* $S * S$ *with respect to Neumann boundary conditions. Then*

$$\begin{aligned}
(D^{(d)})^2 &:= (V_{n_d} \otimes \cdots \otimes V_{n_1})^\top\, A^{(\text{curv},d)}\, (V_{n_d} \otimes \cdots \otimes V_{n_1}) \\
&= \operatorname{diag}((d_{j_1,\dots,j_d})^2,\ j_q = 1, \dots, n_q,\ q = 1, \dots, d)
\end{aligned}$$

where V_m *is defined by eqn (11.10) and* $d_{j_1,\dots,j_d}$ *is defined recursively by*

$$\begin{aligned}
d_{j_1}^{p_{d-1},\dots,p_1} &= S_{2,p_{d-1},\dots,p_1} + 2S_{2,p_{d-1},\dots,p_1} \cos\tfrac{j_1\pi}{n_1}, \\
d_{j_1,\dots,j_k}^{p_{d-k},\dots,p_1} &= d_{j_1,\dots,j_k}^{2,p_{d-k},\dots,p_1} + 2d_{j_1,\dots,j_k}^{1,p_{d-k},\dots,p_1} \cos\tfrac{j_k\pi}{n_k}, \\
d_{j_1,\dots,j_d} &= d_{j_1,\dots,j_{d-1}}^2 + 2d_{j_1,\dots,j_{d-1}}^1 \cos\tfrac{j_d\pi}{n_d}.
\end{aligned}$$

The matrix stencils $S^{\text{curv},d} = S^{\text{diff},d} * S^{\text{diff},d}$ for dimension $d = 2, 3$ are summarize in Table 12.1; for $S^{\text{diff},d}$ see Table 11.1.

As a further by-product of the analysis in Section 11.2, we note that an $\mathcal{O}(n \log n)$ implementation based on the discrete cosine transformation can be deduced. Here n denotes the number of voxels.

Table 12.1 *Matrix stencils* $S^{\text{curv},d} = S^{\text{diff},d} * S^{\text{diff},d}$ *for the bi-harmonic operator and dimension* $d = 2, 3$; $S^{\text{diff},d}$ *are given in Table 11.1.*

$$S^{\text{curv},2} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & -8 & 2 & 0 \\ 1 & -8 & 20 & -8 & 1 \\ 0 & 2 & -8 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix},$$

$$S^{\text{curv},3} = \left(S^{\text{curv},3}_{j,k,\ell}\right)_{j,k,\ell=1,\dots,5}$$

$$= \begin{cases} 42, & j{=}k{=}\ell{=}2, \\ -12, & |j{-}3|{+}|k{-}3|{+}|\ell{-}3|{=}1, \\ 2, & |j{-}3|{+}|k{-}3|{+}|\ell{-}3|{=}2 \\ & \wedge \max\{|j{-}3|{+}|k{-}3|{+}|\ell{-}3|\}{=}1, \\ 1, & |j{-}3|{+}|k{-}3|{+}|\ell{-}3|{=}2 \\ & \wedge \max\{|j{-}3|{+}|k{-}3|{+}|\ell{-}3|\}{=}2, \\ 0, & \text{otherwise.} \end{cases}$$

In contrast to diffusion registration, where even an $\mathcal{O}(n)$ algorithm can be obtained by exploiting an additive operator splitting scheme for a time marching-based implementation, see Section 11.3, curvature registration does not permit such a fast numerical treatment.

This can already be seen for dimension $d = 2$, where the discrete Euler–Lagrange equations (cf., eqn (12.3)) read

$$\frac{\vec{V}^{(k+1)} - \vec{V}^{(k)}}{\tau} + (A_1 + A_2)^2 \vec{V}^{(k+1)} = \vec{F}_j^{(k)},$$

where $\vec{V}^{(k)}$ and $\vec{F}_j^{(k)}$ are defined in eqns (11.15) and (11.16), respectively. The matrices $A_1, A_2 \in \mathbb{R}^{n \times n}$ are given by

$$A_1 = \alpha\, I_{n_2} \otimes M_{n_1}, \quad A_2 = \alpha\, M_{n_2} \otimes I_{n_1},$$

where $n := n_1 n_2$ is the number of pixels, and M_m is defined in eqn (11.5). Using these particular matrices, an AOS scheme for curvature registration reads

$$\vec{V}^{(k+1)} = \frac{1}{3} \sum_{\ell=1}^{3} \vec{V}_\ell^{(k+1)},$$

where

$$\left(I_n + 3\tau I_{n_2} \otimes M_{n_1}^2\right) \quad \vec{V}_1^{(k+1)} = \vec{V}^{(k)} + \tau \vec{F}_j^{(k)}, \tag{12.4}$$

$$\left(I_n + 6\tau M_{n_2} \otimes M_{n_1}\right) \quad \vec{V}_2^{(k+1)} = \vec{V}^{(k)} + \tau \vec{F}_j^{(k)}, \tag{12.5}$$

$$\left(I_n + 3\tau M_{n_2}^2 \otimes I_{n_1}\right) \quad \vec{V}_3^{(k+1)} = \vec{V}^{(k)} + \tau \vec{F}_j^{(k)}. \tag{12.6}$$

For the solution of eqns (12.4) and (12.6), an $\mathcal{O}(n)$ algorithm is available. However, the overall complexity is governed by the solution of eqn (12.5), where only $\mathcal{O}(n \log n)$ direct solution schemes are known. Thus, our implementation is directly based on the factorization presented in Corollary 12.2.

12.2 A relation to thin plate splines

In Chapter 4, the bi-linear form

$$a_q^{\mathrm{TPS}}[\xi, \eta] = \sum_{|\kappa|=q} c_\kappa \int_\Omega D^\kappa \xi\, D^\kappa \eta \, dx$$

was investigated; see, Section 4.3.1. In fact, there is a strong relation between this spline approach and our curvature registration. In the spline approach, one is looking for a u_ℓ which interpolates or approximates certain data but is as smooth as possible. Smoothness is measured in terms of $a_q^{\mathrm{TPS}}[u_\ell, u_\ell]$, which might also be viewed as a curvature.

From the proof of Theorem 12.1, we have

$$\begin{aligned} a^{\mathrm{curv}}[\xi, \eta] &= \int_\Omega \Delta\xi \Delta\eta \, dx \\ &= \int_{\partial\Omega} \Delta\xi \left\langle \nabla\eta, n \right\rangle_{\mathbb{R}^d} - \eta \left\langle \nabla\Delta\xi, n \right\rangle_{\mathbb{R}^d} dx + \int_\Omega \eta \Delta^2 \xi \, dx. \end{aligned}$$

In particular for $q = 2$, we find

$$\begin{aligned} a_2^{\mathrm{TPS}}[\xi, \eta] &= \int_\Omega \partial_{x_1 x_1}\xi \partial_{x_1 x_1}\eta + 2\partial_{x_1 x_2}\xi \partial_{x_1 x_2}\eta + \partial_{x_2 x_2}\xi \partial_{x_2 x_2}\eta \, dx \\ &= \int_\Omega \left\langle \nabla\partial_{x_1}\xi, \nabla\partial_{x_1}\eta \right\rangle_{\mathbb{R}^d} + \left\langle \nabla\partial_{x_2}\xi, \nabla\partial_{x_2}\eta \right\rangle_{\mathbb{R}^d} dx \\ &= \int_{\partial\Omega} \partial_{x_1}\eta \left\langle \nabla\partial_{x_1}\xi, \vec{n} \right\rangle_{\mathbb{R}^d} + \partial_{x_2}\eta \left\langle \nabla\partial_{x_2}\xi, \vec{n} \right\rangle_{\mathbb{R}^d} dx \\ &\quad - \int_\Omega \partial_{x_1}\eta \Delta[\partial_{x_1}\xi] + \partial_{x_2}\eta \Delta[\partial_{x_2}\xi] \, dx \\ &= \int_{\partial\Omega} \nabla\eta [\nabla^2 \xi] \vec{n} \, dx - \int_\Omega \left\langle \nabla\Delta\xi, \nabla\eta \right\rangle dx \\ &= \int_{\partial\Omega} \nabla\eta [\nabla^2 \xi] \vec{n} - \eta \left\langle \nabla\Delta\xi, \vec{n} \right\rangle_{\mathbb{R}^d} dx + \int_\Omega \eta \Delta^2 \xi \, dx. \end{aligned}$$

It is worthwhile noticing that the derivatives of the two functionals $a^{\mathrm{curv}}[\xi, \xi]$ and $a_2^{\mathrm{TPS}}[\xi, \xi]$ share the same main part.

12.3 Curvature registration: squares

We begin with a small academic example, given the reference and template images depicted in Fig. 12.1. We perform both curvature and fluid registration. In particular we choose $\alpha = 10^6$, $\tau = 5$, $\mu = 10^3$, and $\lambda = 0$. The deformed grids are also illustrated in Fig. 12.1. Intermediate results are depicted in Fig. 12.2.

Both curvature and fluid registration give meaningful registration results. For both techniques, the deformed template images are (besides some interpolation artifacts) identical to the reference image, and the deformation is visually pleasing. However, the deformations are completely different. For curvature registration we detect an almost linear deformation of the original image. Note that affine linear rergistrations are not penalized by the curvature regularizer. On the other hand, for fluid registration, the deformation is not linear at all.

This example also illustrates that the quality of registration cannot be measured by inspecting the deformed template or the difference image. Moreover, for this particular example, it is not possible to decide which of the two registrations

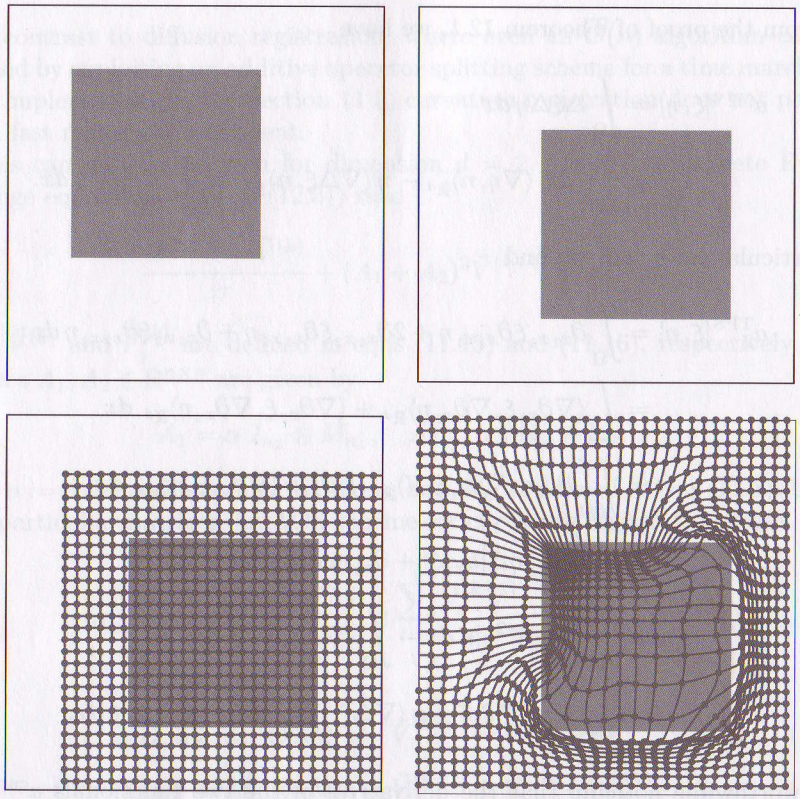

FIG. 12.1 TOP LEFT: reference image, TOP RIGHT: template image, BOTTOM LEFT: deformed grid after curvature registration ($\alpha = 10^6$ and $\tau = 5$), and BOTTOM RIGHT: deformed grid after fluid registration ($\mu = 10^3$ and $\lambda = 0$).

is better. The problem is that image registration is an ill-posed problem. The distance measure can have two different meaningful solutions, the regularizer term deciding which one is more likely. One topic of regularization is to distinguish a likely solution on purpose. Note that this ambiguity is intrinsic to the problem and has nothing to do with "complicated" regularizing terms. Even when restricted to the class of rigid transformations, the actual example has multiple solutions, e.g., the particular translation and a rotation of about 180 degrees.

12.4 Curvature registration: • to C

We continue the registration of a • to a C, see also Section 9.8, Section 10.5, and Section 11.4. In Fig. 12.3, we show the intermediate results of curvature registration for $k = 0(20)100$. In order to visualize the deformation, the template

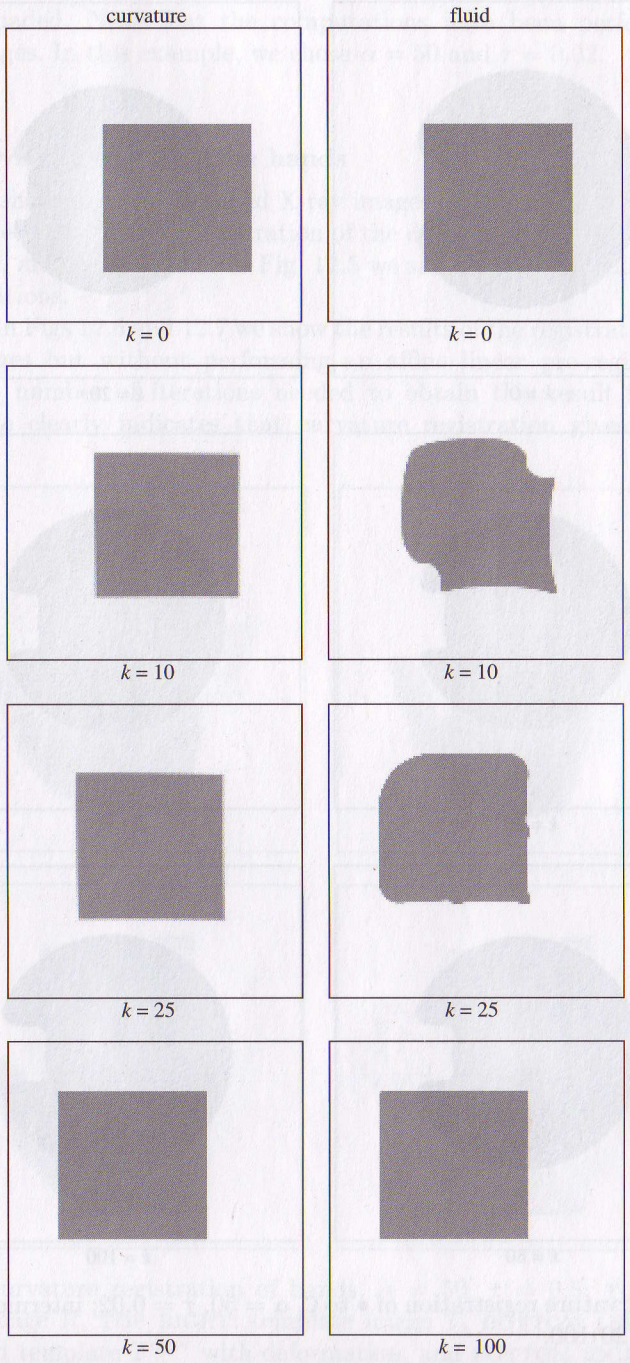

FIG. 12.2 Curvature (LEFT) and fluid (RIGHT) registration of squares: intermediate results.

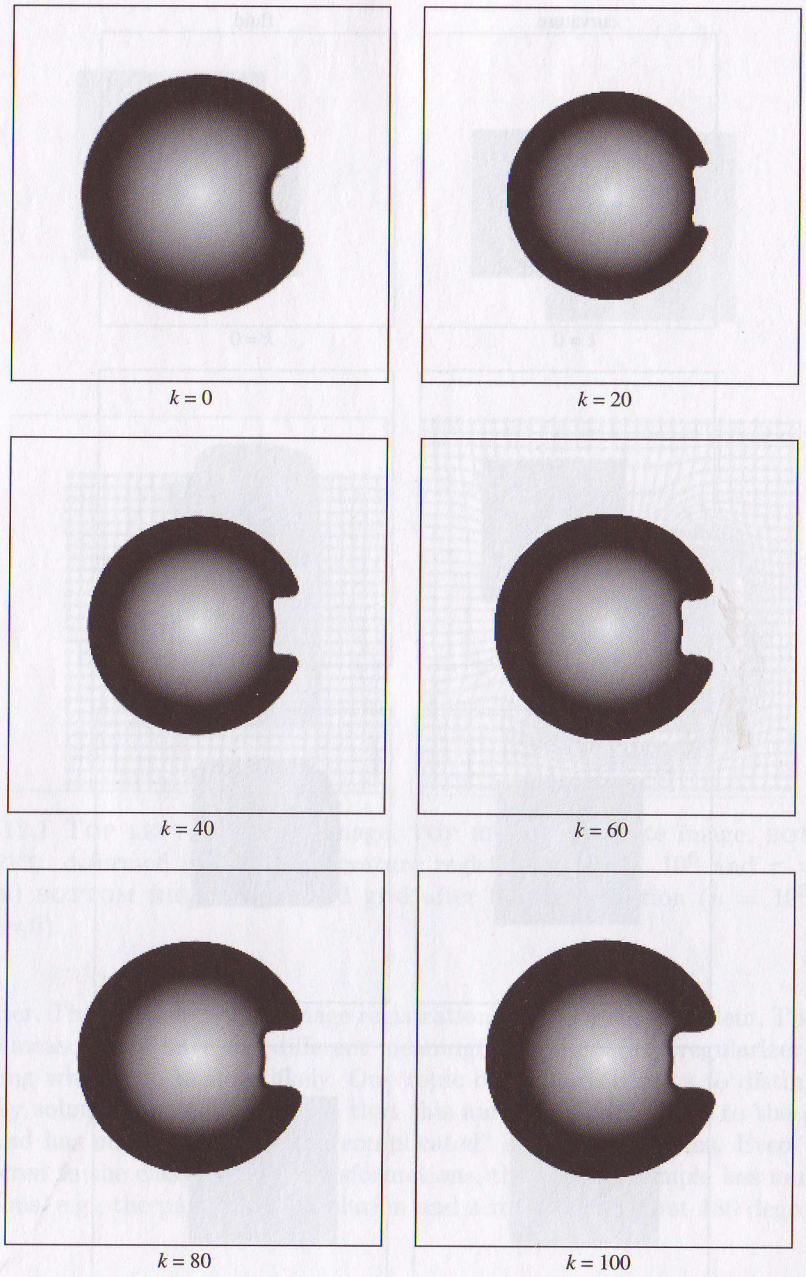

FIG. 12.3 Curvature registration of • to C, $\alpha = 50$, $\tau = 0.02$; intermediate results for $k = 0(20)100$.

has been shaded. Note that the computations have been performed on the original images. In this example, we chose $\alpha = 50$ and $\tau = 0.02$.

12.5 Curvature registration: hands

Figure 12.4 shows the two modified X-ray images of human hands and the diffusion registered hand with an illustration of the deformation, see also Section 9.9, Section 10.6, and Section 11.5. In Fig. 12.5 we show the intermediate results for various iterations.

Finally, in Figs 12.6 and 12.7 we show the results of the registration performed on the images but without performing an affine linear pre-registration step. Though the number of iterations needed to obtain this result is quite high, this example clearly indicates that curvature registration gives much better

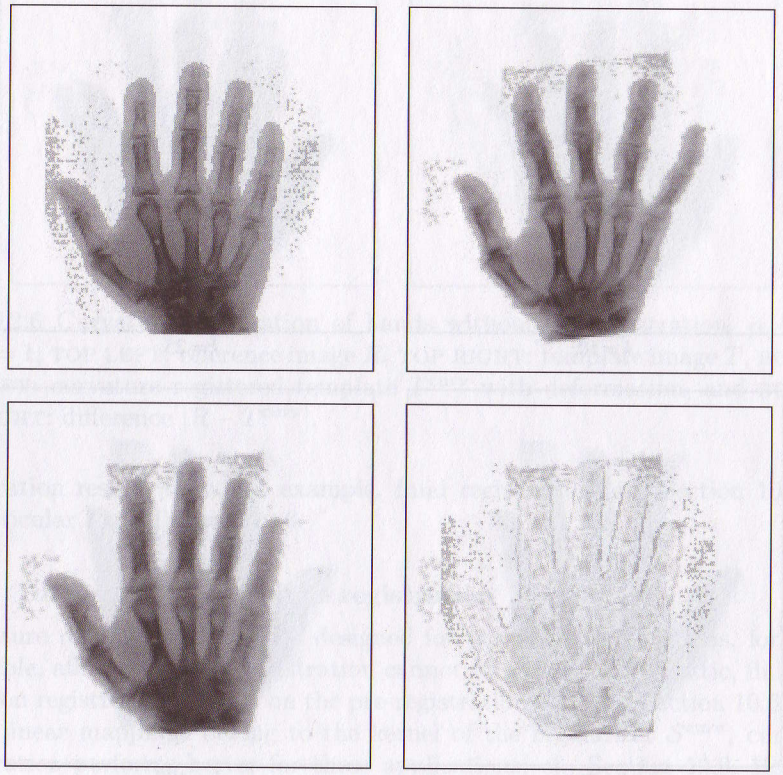

FIG. 12.4 Curvature registration of hands, $\alpha = 50$, $\tau = 0.5$; TOP LEFT: reference image R, TOP RIGHT: template image T, BOTTOM LEFT: curvature registered template T^{curv} with deformation, and BOTTOM RIGHT: difference $|R - T^{\text{curv}}|$.

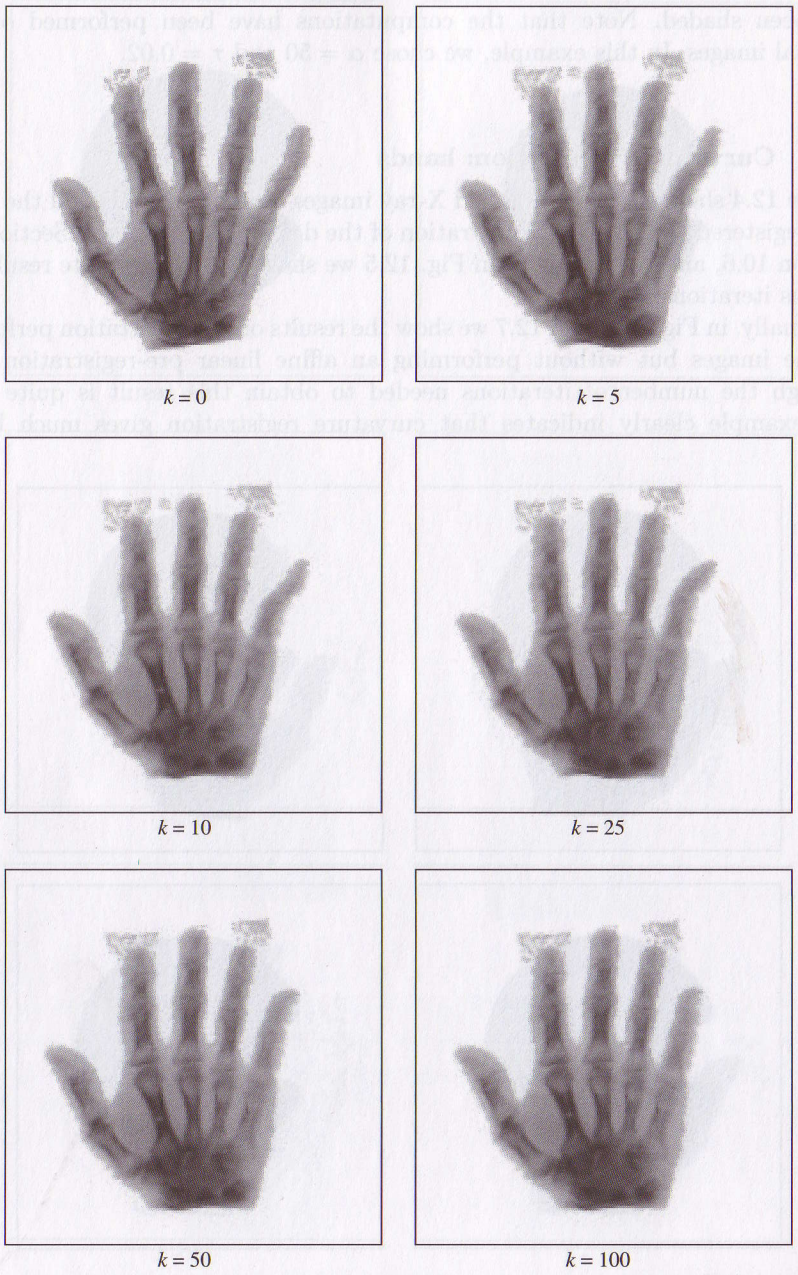

FIG. 12.5 Curvature registration of hands ($\alpha = 50$, $\tau = 0.5$), intermediate results for $k = 0, 5, 10, 25, 50, 100$.

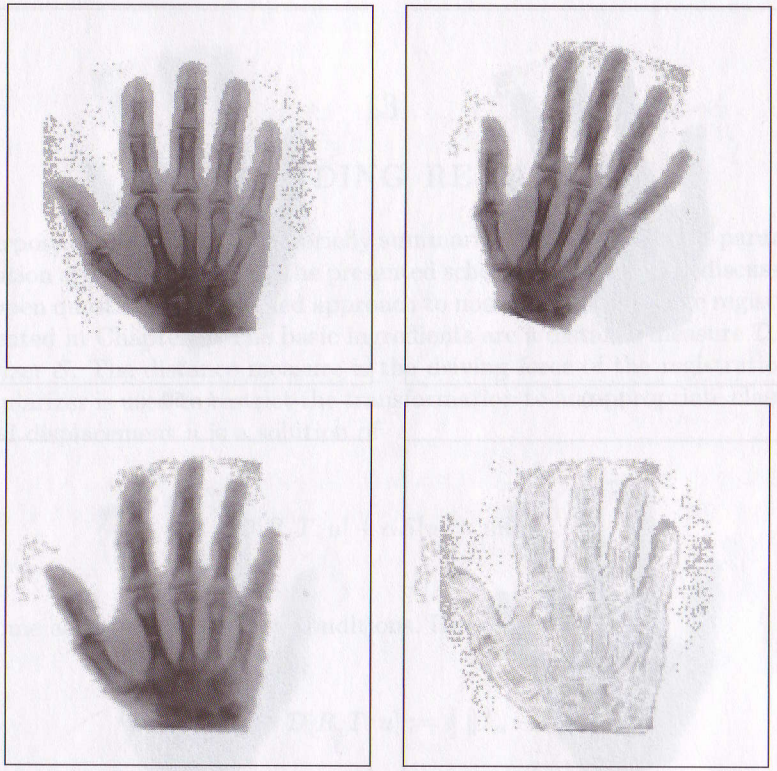

FIG. 12.6 Curvature registration of hands without pre-registration, $\alpha = 10^5$, $\tau = 1$; TOP LEFT: reference image R, TOP RIGHT: template image T, BOTTOM LEFT: curvature registered template T^{curv} with deformation, and BOTTOM RIGHT: difference $|R - T^{\text{curv}}|$.

registration results than, for example, fluid registration; cf., Section 10.6 and in particular Figs 10.5 and 10.6.

12.6 Discussion of curvature registration

Curvature registration has been designed for particular applications, for which a reliable, affine linear pre-registration cannot be guaranteed. Elastic, fluid, and diffusion registration depend on the pre-registration step; cf., Section 10.6. Since affine linear mappings belong to the kernel of the regularizer $\mathcal{S}^{\text{curv}}$, curvature registration performs better for these applications; cf., Section 12.3. However, the initial position plays an important role also for curvature registration and it is not advisable to skip the pre-registration.

The curvature registration regularizer is based on second order derivatives. Thus, the transformation is smoother than the ones obtained by a first order derivatives regularizer, like, e.g., diffusion registration.

FIG. 12.7 Curvature registration of hands without pre-registration ($\alpha = 50$, $\tau = 0.5$), intermediate results for $k = 0, 50, 100, 250, 500, 1000$.

13

CONCLUDING REMARKS

The purpose of this chapter is to briefly summarize the proposed non-parametric registration approach as well as the presented schemes and to briefly discuss some of the open questions. The unified approach to non-parametric image registration is presented in Chapter 8. The basic ingredients are a distance measure $\mathcal{D}$ and a regularizer $\mathcal{S}$. The distance measure is the driving force of the registration and the regularizer is used to restrict the transformation to an appropriate class. The required displacement u is a solution of

$$\mathcal{D}[R, T; u] + \alpha \mathcal{S}[u] = \min,$$

with some additional boundary conditions. Here, we used

$$\mathcal{D}[R, T; u] := \mathcal{D}[R, T; u] := \tfrac{1}{2} \left\| T_u - R \right\|_{L_2(\Omega)};$$

cf., Definition 6.1 (see also Theorem 8.1). For the regularizer $\mathcal{S}$, we discussed four particular choices which then lead to the elastic, fluid, diffusion, and curvature registration schemes.

The numerical treatment presented is based on the corresponding Euler–Lagrange equations, which lead to a system of non-linear partial differential equations, $\mathcal{A}[u] = f(\,\cdot\,, u)$; cf., eqn (8.5). A fixed-point or time marching algorithm is used in order to circumvent the non-linearity of f with respect to u. After appropriate finite difference approximations of the time-discrete equations, we ended up with an iterative scheme, where in each step a large system of linear equation has to be solved. Particular solution techniques are designed to reduce the numerical complexity of the overall iterative schemes.

In Section 13.1, we briefly summarize the elastic, fluid, diffusion, and curvature registration schemes. In Section 13.2, we comment on timings obtained in a particular environment. A remark on how to compare the different approaches is given in Section 13.4. We also comment on the choice of the parameters λ, μ and α, τ, respectively (cf., Section 13.5), on further acceleration techniques of the schemes (cf., Section 13.6), and on extensions (cf., Section 13.7).