MATCHING DEFORMABLE MODELS TO IMAGES: DIRECT AND ITERATIVE SOLUTIONS

Demetri Terzopoulos

Schlumberger Palo Alto Research
3340 Hillview Avenue, Palo Alto, CA 94304

In a companion paper [7] we formulate a broadly applicable class of mathematical models for describing the shapes of objects appearing in images. These models may be viewed as deformable bodies composed of an abstract elastic material. They are matched to images through the action of externally applied forces derived from image functions. Additional guiding forces may be applied interactively by the user. Three applications have served to illustrate the utility of the models: image based contour extraction, correspondence matching, and 3D object reconstruction. In the present paper, we deal with the discrete solution of the dynamical equations inherent to these applications. We focus on the direct and iterative numerical methods underlying our implemented algorithms for matching deformable models to images.

Lagrangian Dynamics of Deformable Models

The variational principles defined in [7] characterize the desired equilibrium configurations of the deformable models. When the external forces change, potential energy is converted to kinetic energy, the equilibrium is destroyed, and the models dynamically deform away from their prescribed natural states. If kinetic energy is dissipated by damping, then the transients decay and a new equilibrium is reached. Given the deformation potential energy functional $\mathcal{E}_q(\mathbf{v})$ defined by equation (1) (equations (1)–(4) are found in [7]), we define the Lagrangian functional

$$\mathcal{L}(\mathbf{v}) = \frac{1}{2}\int_\Omega \mu(\mathbf{x})|\mathbf{v}_t|^2\,d\mathbf{x} - \frac{1}{2}\mathcal{E}_q(\mathbf{v}), \tag{5}$$

as well as the (Rayleigh) dissipation functional $\mathcal{D}(\mathbf{v}_t) = \frac{1}{2}\int_\Omega \gamma(\mathbf{x})|\mathbf{v}_t|^2\,d\mathbf{x}$, where $\mu(\mathbf{x})$ is the mass density, $\gamma(\mathbf{x})$ is the damping density, and the subscript t denotes a time derivative.

If the initial configuration is $\mathbf{u}(\mathbf{x},t_0)$, the deformable model's motion $\mathbf{u}(\mathbf{x},t)$ is such that

$$\frac{\delta}{\delta u_i}\left(\int_{t_0}^{t_1} \mathcal{L}(\mathbf{u}) + \mathcal{D}(\mathbf{u}_t)\,dt\right) = 0, \qquad i = 1,\ldots,d. \tag{6}$$

This condition, that the action integral is stationary, leads to Lagrange's general equations of motion [8]. For our deformable model, it can be shown that they are given by

$$\frac{\partial}{\partial t}\left(\mu\frac{\partial u_i}{\partial t}\right) + \gamma\frac{\partial u_i}{\partial t} + \sum_{m=0}^{q}(-1)^m\Delta^m_{\mathbf{w}_m}u_i = -\frac{\partial P}{\partial u_i}, \qquad i = 1,\ldots,d, \tag{7}$$

where $\Delta^m_{\mathbf{w}_m}$ is the weighted iterated Laplacian operator defined by

$$\Delta^m_{\mathbf{w}_m} = \sum_{j_1+\cdots+j_p=m}\frac{m!}{j_1!\ldots j_d!}\,\frac{\partial^m}{\partial x_1^{j_1}\ldots\partial x_d^{j_d}}\left(w_j(\mathbf{x})\frac{\partial^m}{\partial x_1^{j_1}\ldots\partial x_d^{j_d}}\right) \tag{8}$$

and $j = (j_1,\ldots,j_p)$ is a multi-index. Associated with these equations are appropriate initial and boundary conditions.

Discretization and Solution Methods

A general approach to discretizing the equations of motion is to represent admissible configurations in the form of a linear superposition of basis functions. Discontinuities in the model's elastic properties are most conveniently dealt with through the use of local-support basis functions. We

employ the finite element method [9], a systematic approach to obtaining piecewise solutions in finite element subspaces of the admissible space $\mathcal{H}$. The basis functions of finite element subspaces are (typically) low-order polynomials. The number of basis functions depends on the number of finite elements employed to tessellate the spatial domain Ω. The tessellation can be irregular, if necessary. Alternatively the finite difference method [9] can be used to discretize the Lagrange equations of motion. In practice we employ aspects of both of these complementary methods; e.g., finite elements in space and finite differences in time.

To compute the dynamics, we have employed explicit and implicit Euler methods [10] for numerically integrating the second-order Lagrange equations (7). Implicit methods require the solution of a time sequence of discrete equilibrium subproblems. The subproblem at time t may generally be expressed as a coupled set of d positive definite, symmetric systems of algebraic equations

$$\mathbf{A}_k^t \mathbf{u}_k^t = \mathbf{g}_k^t, \qquad k = 1, \ldots, d \tag{9}$$

that must be solved for the unknown instantaneous configurations $\mathbf{u}^t$. The size of the linear systems depends on the number of points in the discrete model. Although the $\mathbf{A}_i$ can be large for higher-dimensional models, they will be sparse due to the local support of the basis functions.

Either direct or iterative sparse system solution methods are applicable [9, 10]. On serial computers, direct methods will lead to linear-time complexity algorithms for the $p = 1$ parameter case. In general, iterative methods are better suited for massively parallel machines, and they become especially advantageous when $p > 1$. Multigrid relaxation methods can provide nearly optimal convergence rates for large multidimensional matching problems [1].

A Direct Method for the Case $p = 1$

The deformable contour or "snake" model is a one-parameter model whose potential energy of deformation is given by (2). Assuming constant mass density $\mu(s) = \mu$ and constant dissipation $\gamma(s) = \gamma$, the associated Lagrange dynamical equations (7) reduce to

$$\mu \mathbf{u}_{tt} + \gamma \mathbf{u}_t + \frac{\partial}{\partial s}\left(w_1 \mathbf{u}_s\right) + \frac{\partial^2}{\partial s^2}\left(w_2 \mathbf{u}_{ss}\right) = -\nabla P(\mathbf{u}(s,t)), \qquad \mathbf{u}(s) = (x(s), y(s)). \tag{10}$$

This equation is satisfied in $\Omega = (0,1)$ with prescribed boundary conditions on $\partial\Omega = \{0,1\}$. We regularly tessellate Ω to obtain the nodal configuration variables $\mathbf{u}_i = \mathbf{u}(ih)$, where $h = 1/N$ and N is the number of nodes.

Given a time step Δt, the backward difference expressions $a\mu(\mathbf{u}_i^t - 2\mathbf{u}_i^{t-1} + \mathbf{u}_i^{t-2})$ and $b\gamma(\mathbf{u}_i^t - \mathbf{u}_i^{t-2})$, where $a = 1/(\Delta t)^2$ and $b = 1/2\Delta t$, may be used to obtain an (semi) implicit Euler formula in time. This formula can be expressed in the matrix form (9) as

$$\mathbf{A}\mathbf{x}^t = \mathbf{g}_x^t, \qquad \mathbf{A}\mathbf{y}^t = \mathbf{g}_y^t, \tag{11}$$

where $\mathbf{x}^t = [x_0^t, \ldots, x_N^t]$ and $\mathbf{y}^t = [y_0^t, \ldots, y_N^t]$ are the (unknown) nodal position vectors at time t. The vectors on the right hand side, which depend on two prior configurations, are given by $\mathbf{g}_x^t = (a\mu + b\gamma)\mathbf{x}^{t-1} + (a\mu - b\gamma)(\mathbf{x}^{t-1} - \mathbf{x}^{t-2}) - \mathbf{f}_x$ and $\mathbf{g}_y^t = (a\mu + b\gamma)\mathbf{y}^{t-1} + (a\mu - b\gamma)(\mathbf{y}^{t-1} - \mathbf{y}^{t-2}) - \mathbf{f}_y$, where the generalized force vector components contain partial derivatives of P: $\mathbf{f}_x = [P_x(x_0^{t-1}, y_0^{t-1}), \ldots, P_x(x_N^{t-1}, y_N^{t-1})]$ and $\mathbf{f}_y = [P_y(x_0^{t-1}, y_0^{t-1}), \ldots, P_y(x_N^{t-1}, y_N^{t-1})]$.

We write $\mathbf{A} = (a\mu - b\gamma)\mathbf{I} + \mathbf{K}$, where $\mathbf{I}$ is a unit matrix and $\mathbf{K}$ is the stiffness matrix associated with the internal deformation energy of the material. It is common practice in the finite element method to automatically assemble the stiffness matrix. Quadratic finite elements can be employed to discretize the spatial derivatives in the energy functional (2) or, alternatively, the spatial derivatives in (10) may be approximated using the standard $O(h^2)$ central finite differences.

For cyclic boundary conditions (i.e., a closed contour), the stiffness matrix is the pentadiagonal

symmetric matrix shown below (unspecified entries are 0):

$$\mathbf{K} = \begin{pmatrix} c_0 & b_0 & a_0 & & & & & a_{N-1} & b_N \\ b_0 & c_1 & b_1 & a_1 & & & & & a_N \\ a_0 & b_1 & c_2 & b_2 & a_2 & & & & \\ & a_1 & b_2 & c_3 & b_3 & a_3 & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & a_{N-4} & b_{N-3} & c_{N-2} & b_{N-2} & a_{N-2} & \\ a_{N-1} & & & & a_{N-3} & b_{N-2} & c_{N-1} & b_{N-1} & \\ b_N & a_N & & & & a_{N-2} & b_{N-1} & c_N & \end{pmatrix}, \tag{12}$$

where $c_i = 2w_1(ih) + 6w_2(ih)$, $b_i = -w_1(ih) - 4w_2(ih)$, and $a_i = w_2(ih)$. To insert a position discontinuity at $s = (i - 1/2)h$ between nodes $i-1$ and i along the contour, we use the technique of computational molecules [11], which dictates that the following 9 entries be modified: $c_i = w_1(ih) + w_2(ih)$, $c_{i-1} = w_1(ih - h) + w_2(ih - h)$, $c_{i+1} = 2w_1(ih + h) + 5w_2(ih + h)$, $c_{i-2} = 2w_1(ih - 2h) + 5w_2(ih - 2h)$, $b_i = -w_1(ih) - 2w_2(ih)$, $b_{i-2} = -w_1(ih - 2h) - 2w_2(ih - 2h)$, and $a_{i-1} = b_{i-1} = a_{i-2} = 0$. Similarly, one way of inserting a tangent discontinuity at $s = ih$ is to make the modifications $c_i = 2w_1(ih) + 2w_2(ih)$, $c_{i-1} = 2w_1(ih - h) + 5w_2(ih - h)$, $c_{i+1} = 2w_1(ih + h) + 5w_2(ih + h)$, $b_i = -w_1(ih) - 2w_2(ih)$, $b_{i-1} = -w_1(ih - h) - 2w_2(ih - h)$, $a_{i-1} = 0$.

The pentadiagonal system can be solved very efficiently ($O(N)$ time and space) by factorizing $\mathbf{A}$ into lower and upper triangular matrices, then solving the two resulting sparse triangular systems [10]. We compute the unique normalized factorization $\mathbf{A} = \mathbf{DLUD}$ where $\mathbf{D}$ is a diagonal matrix, $\mathbf{L}$ is a lower triangular matrix, and $\mathbf{U} = \mathbf{L}^\top$ [12]. The solution $\mathbf{x}^t = \mathbf{D}^{-1}\mathbf{r}$ to (11) is obtained by first solving $\mathbf{Lz} = \mathbf{D}^{-1}\mathbf{g}_x^t$ by forward substitution, then $\mathbf{Ur} = \mathbf{z}$ by back substitution, and similarly for $\mathbf{y}^t$. Aside from the initial factorization, additional factorizations are required only as $\mathbf{A}$ changes. If $\mathbf{A}$ changes at each time step, it becomes advantageous to perform the factorization simultaneously with the forward substitution. Note that the factorization and substitutions are recursive and inherently sequential operations.

Iterative Methods for Cases $p > 1$

Consider the correspondence matching problem ($p = 2$) which involves the deformation energy (3), where $\mathbf{u}(x, y, t) = (u_1(x, y, t), u_2(x, y, t))$. Assuming constant mass density and dissipation, the Lagrange dynamical equations (7) reduce in Ω to

$$\mu\mathbf{u}_{tt} + \gamma\mathbf{u}_t - \frac{\partial}{\partial x}(w_1\mathbf{u}_x) - \frac{\partial}{\partial y}(w_1\mathbf{u}_y) + \frac{\partial}{\partial x^2}(w_2\mathbf{u}_{xx}) + 2\frac{\partial}{\partial x\partial y}(w_2\mathbf{u}_{xy}) + \frac{\partial}{\partial y^2}(w_2\mathbf{u}_{yy}) = -\nabla P(\mathbf{u}(x, y, t)), \tag{13}$$

with appropriate boundary conditions on $\partial\Omega$ and initial conditions at $t = t_0$, Discretization of the horizontal and vertical disparity functions in space leads to the vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ of nodal variables. If, as before, an implicit Euler formula is set up by taking backward time differences, there results a sequence of systems in the form of (11) but with $\mathbf{u}_1$ and $\mathbf{u}_2$ playing the role of $\mathbf{x}$ and $\mathbf{y}$.

Although the positive definite, symmetric matrix $\mathbf{A}$ remains sparse and banded when $p > 1$, bands will occur at substantial distances from the main diagonal. In particular, if the two-parameter disparity model (3) or deformable shell model (4) are discretized on an $M \times N$ grid using the 13-point (biharmonic) computational molecules (see [1]), diagonal bands are introduced at a distance $N - 1$, N, $N + 1$, and $2N$ away from the main diagonal. Unfortunately, factorization destroys nearly all zero entries within the bandwidth $2N$. The resulting $O(MN^2)$ time and space complexities of obtaining a solution are severe for large problems.

Iterative methods with the property that no nonzero entries are introduced are advantageous for higher-dimensional problems. The general stationary iterative method is obtained by writing (9) in the form $\mathbf{u}^{(k+1)} = \mathbf{Gu}^{(k)} + \mathbf{c}$ where $\mathbf{G}$ is a fixed iteration matrix and where the bracketed superscripts denote the iteration number. There are several practical variants. A gradient relaxation

method is defined by the iteration matrix $\mathbf{G} = \mathbf{I} - \alpha\mathbf{A}$ and $\mathbf{c} = \alpha\mathbf{g}$, where α is a step size. Writing $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$, the sum of a diagonal, a strictly lower triangular, and a strictly upper triangular matrix (these matrices differ from those in the factorization method), the successive over-relaxation method with relaxation parameter ω is defined by $\mathbf{G} = (\mathbf{I} - \omega\mathbf{D}^{-1}\mathbf{L})^{-1}[(1-\omega)\mathbf{I} + \omega\mathbf{D}^{-1}\mathbf{U}]$ and $\mathbf{c} = (\mathbf{I} - \omega\mathbf{D}^{-1}\mathbf{L})^{-1}\omega\mathbf{D}^{-1}\mathbf{g}$.

To illustrate the iterative solution of the problem we set $\mu = 0$ and take a forward difference in time to obtain a simple, explicit Euler formula. Equilibrium is achieved in the limit as $t \to \infty$. The two-parameter spatial domain Ω is regularly tessellated into squares of size h, and the energy functional (4) may be discretized using quadratic finite elements [1, 11] or the spatial derivatives in (13) may be discretized using central differences. The resulting gradient relaxation process is

$$\begin{aligned} u_{1\,i,j}^{(k+1)} &= u_{1\,i,j}^{(k)} + (\Delta t/\gamma)\big(\Phi_{i,j}(\mathbf{u}_1^{(k)}) + P_{u_1}(u_{1\,i,j}^{(k)}, u_{2\,i,j}^{(k)})\big), \\ u_{2\,i,j}^{(k+1)} &= u_{2\,i,j}^{(k)} + (\Delta t/\gamma)\big(\Phi_{i,j}(\mathbf{u}_2^{(k)}) + P_{u_2}(u_{1\,i,j}^{(k)}, u_{2\,i,j}^{(k)})\big), \end{aligned} \tag{14}$$

where $\Phi_{i,j}(\mathbf{u}) = -w_{1\,i,j}(\mathrm{u}_{i+1,j} - \mathrm{u}_{i,j}) + w_{1\,i-1,j}(\mathrm{u}_{i,j} - \mathrm{u}_{i-1,j}) - w_{1\,i,j}(\mathrm{u}_{i,j+1} - \mathrm{u}_{i,j}) + w_{1\,i,j-1}(\mathrm{u}_{i,j} - \mathrm{u}_{i,j-1}) + (1/h^2)\{w_{2\,i-1,j}(\mathrm{u}_{i,j} - 2\mathrm{u}_{i-1,j} + \mathrm{u}_{i-2,j}) - 2w_{2\,i,j}(\mathrm{u}_{i+1,j} - 2\mathrm{u}_{i,j} + \mathrm{u}_{i-1,j}) + w_{2\,i+1,j}(\mathrm{u}_{i+2,j} - 2\mathrm{u}_{i+1,j} + \mathrm{u}_{i,j}) + 2w_{2\,i-1,j-1}(\mathrm{u}_{i,j} - \mathrm{u}_{i-1,j} - \mathrm{u}_{i,j-1} + \mathrm{u}_{i-1,j-1}) - 2w_{2\,i,j-1}(\mathrm{u}_{i+1,j} - \mathrm{u}_{i,j} - \mathrm{u}_{i+1,j-1} + \mathrm{u}_{i,j-1}) - 2w_{2\,i-1,j}(\mathrm{u}_{i,j+1} - \mathrm{u}_{i-1,j+1} - \mathrm{u}_{i,j} + \mathrm{u}_{i-1,j}) + 2w_{2\,i,j}(\mathrm{u}_{i+1,j+1} - \mathrm{u}_{i,j+1} - \mathrm{u}_{i+1,j} + \mathrm{u}_{i,j}) + w_{2\,i,j-1}(\mathrm{u}_{i,j} - 2\mathrm{u}_{i,j-1} + \mathrm{u}_{i,j-2}) - 2w_{2\,i,j}(\mathrm{u}_{i,j+1} - 2\mathrm{u}_{i,j} + \mathrm{u}_{i,j-1}) + w_{2\,i,j+1}(\mathrm{u}_{i,j+2} - 2\mathrm{u}_{i,j+1} + \mathrm{u}_{i,j})\}$ in Ω. Other expressions are obtained at discontinuities, which inhibit computational molecules [11].

Implementation Notes

By expressing (7) as coupled first-order equations in Hamilton's canonical form [8], we have successfully applied standard numerical methods of varying sophistication, other than Euler's method. These include a fourth-order Runge-Kutta method, and Adams-Moulton predictor-corrector methods [10]. The latter offer the advantage of adaptable time-step control, making them particularly robust.

To date, we have implemented four different algorithms for the "snake" matching problem [4], based on successive over-relaxation (author and A. Witkin), two-level multigrid relaxation (J. Platt and author), recursive IIR filtering (M. Kass), and matrix factorization (author). The direct methods, factorization and filtering, are more efficient than the relaxation methods, especially when the rigidity of the deformable contour is high.

The author has implemented iterative algorithms for matching higher-dimensional deformable models to images. The gradient relaxation method described above is used in signal matching as well as stereo and motion matching code [5]. A successive over-relaxation method is used in a program for matching the 3D deformable shell model (4) to images , and the direct method can be used to solve the spine equations [6]. J. Platt has implemented general-purpose multigrid relaxation code which promises to be useful here. Block relaxation [9], a technique combining iteration and matrix factorization, may also offer efficient solutions for $p > 1$.

References

1–6. See the reference list in [7].

7. **Terzopoulos, D.**, "On matching deformable models to images," *These Proceedings.*
8. **Goldstein, H.**, *Classical Mechanics*, Addison-Wesley, Reading, MA, 1965.
9. **Gladwell, I., and Wait, R.**, (ed.), *A Survey of Numerical Methods for Partial Differential Equations*, Clarendon, Oxford, 1979.
10. **Dahlquist, G., and Björck, A.**, *Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1974.
11. **Terzopoulos, D.**, *Proc. 8th Int. J. Conf. AI*, Karlsruhe, FRG, 1983, 1073–1077.
12. **Benson, A., and Evans, D.J.**, *ACM Trans. Mathematical Software*, **3**, 1977, 96–103.