III. 2-d elastic models - image matching

III.1 Introduction

- In images of an object class there is much more information than shape of contour. Images of object class exhibit many topological and geometric similarities which are ignored when the contour alone is computed.
- Variations between images are often smooth - elastic.

  With some pulling, stretching and squeezing - take one image from the class and transform it into another.
- $Z = R$ - The set Z is the entire reference grid.

  Θ - smooth one-to-one mappings of R into L.

  $T(z), z \in R$ template image of the object: typically one random image of the family of images.
- Initial θ given - in particular scale and translation.
- Model $I(x)$ for x only around θR. No background modeling.

  Any image of the object is generated as

$$I(\theta(z)) = T(z) + N(z), \;\; z \in R, \tag{III.1}$$

,

$N(z), z \in Z$ is i.i.d Gaussian.

Pixel intensity of template at z is moved to $\theta(z)$ and noise is added.

Distribution of data I given θ - product distribution as in equation I.2, with only the 'on' object factor present.

- Consider model on continuum and rewrite

$$I(x) = T(\phi(x)) + N(x), \quad \phi = \theta^{-1},$$

for x in a continuous domain D corresponding to R, e.g. the unit square.

- ϕ smooth - extremal points are preserved, smooth level curves mapped into level curves. Topography of T and $T(\phi)$ are the same. See figure III.1.

  Points of interest which are characterized by some peculiar topography of the image intensity surface, are mapped to points with the same local topography and they maintain their relative spatial arrangement.

  If contour of object is marked in the template image T, then ϕ takes it to contour of deformed object in I.

- More complex data models are suggested in Wiskott et al. (1997).

  Local linear operators $J_i, i = 1, \ldots, m$ applied to template and image at each pixel -

  $FI(x), x \in L = (J_1 I(x), \ldots, J_m I(x)), x \in L.$

  $FT(x), x \in L = (J_1 T(x), \ldots, J_m T(x)), x \in L.$

  The model could then be

$$FI_i(x) = FT_i(\phi(x) + N_i(x),$$

where $N_i(x)$ are i.i.d mean zero Gaussians.

Note: in Wiskott et al. (1997) Z is a subset of the reference grid. Their model has some similarities to the model in section V.4.

Figure III.1: Four random deformations of a face

III.2 The variational problem

- Define displacement field: $U(x) = (U^{(1)}(x), U^{(2)}(x)) = \phi(x) - x$.
- Cost function

$$J(U) = \mathcal{E}(U) + \frac{1}{2}\int_{I^2} |T(x + U(x)) - I(x)|^2 dx, \qquad \text{(III.2)}$$

  $\mathcal{E}$ - Smoothing penalty.

- Parameterizations of U

  Spatial - pointwise: $U(x), x \in L$.

  Spectral: U expanded in function basis: Fourier, wavelet, wavelet packet, splines.

$$U^{(j)}(x) = \sum_{k=1}^{\infty} u_k^{(j)} \phi_k(x), j = 1, 2. \qquad \text{(III.3)}$$

  For simplicity assume U is periodic on unit square.

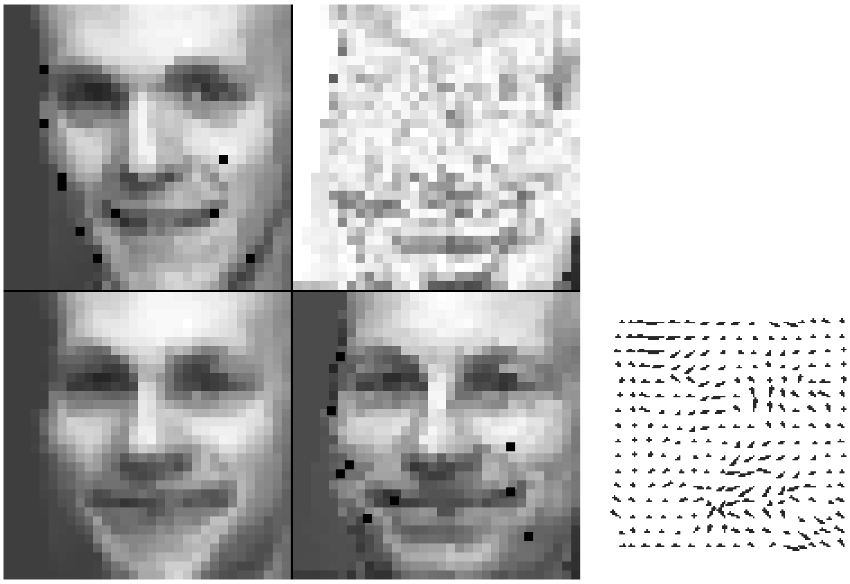

Figure III.2: Top left face serves as T, and is deformed into the bottom left face, $T(\phi)$, when trying to match the bottom right face I. The map ϕ is represented on the right. The dots represent arbitrary locations chosen on the image T and where they are mapped to in the image I. For each point x in the top left observe $\phi^{-1}(x)$ in the bottom right.

- Possible smoothing penalties.

  Spatial - involving derivatives:

$$\mathcal{E}(U) = \int_D \left[\mathcal{L}U^{(1)}(x)\right]^2 + \left[\mathcal{L}U^{(2)}(x)\right]^2 dx. \tag{III.4}$$

  with $\mathcal{L}$ - differential operator.

  Spectral:

$$\mathcal{E}(U) = \sum_{k=1}^{\infty} \alpha_k((u_k^{(1)})^2 + (u_k^{(2)})^2), \tag{III.5}$$

  with α_n some increasing sequence of positive weights.

  Any penalty of the type in equation III.4 can be written as a penalty of the type in equation III.5, when the the Fourier basis is used. Spectral type - richer collection of possible penalties.

- Rewrite cost function $\mathbf{u}$ as

$$J(\mathbf{u}) = \sum_{k=1}^{\infty} \alpha_k ((u_k^{(1)})^2 + (u_k^{(2)})^2) + \tag{III.6}$$
$$\int_D \left(T\left[x_1 + \sum_k u_k^{(1)}\phi_k(x), x_2 + \sum_k u_k^{(2)}\phi_k(x)\right] - I(x)\right)^2 dx.$$

III.2.1 Optimization

- Derivative of the cost function J in the coefficients $u_k^{(j)}$ is given by

$$\frac{\partial J(u^{(1)}, u^{(2)})}{\partial u_k^{(j)}} = \alpha_k u_k^{(j)} + \int \frac{\partial T}{\partial x_j}(x + U(x))$$
$$[T(x + U(x)) - I(x)]\phi_k(x) dx, \tag{III.7}$$

for $i = 1, 2$ and $k = 1, 2 \ldots$ and where U is given by equation III.3.

- Again derivative of the data term is forward transform of

$$T_j(x + U(x))[T(x + U(x) - I(x)],$$

where $T_j = \partial T / x_j,\ j = 1, 2.$

- Ψ - forward transform, Ψ^{-1} - backward transform, with respect to the chosen basis.

Deformation Algorithm:

1. Initialize $\mathbf{u}_0 = \{u_{k,0}^{(j)},\ k = 0, \ldots, d\}, j = 1, 2$. (Typically $\mathbf{u}_0 = 0$). Set $T = 0$.
2. Calculate $U_T^{(1)} = \Psi^{-1}(\mathbf{u}_T^{(1)}), U_T^{(2)} = \Psi^{-1}(\mathbf{u}_T^{(2)})$,
3. Calculate

$$W_j(x) = T_j\Big(x_1 + U_T^{(1)}(x), x_2 + U_T^{(2)}(x)\Big)\Big(T(x_1 + U_T^{(1)}(x), x_2 + U_T^{(2)}(x)) - I(x)\Big),\ j = 1, 2,$$

4. Calculate $\mathbf{v^{(j)}} = \Psi(W_j), j = 1, 2$
5. Set $u^{(j)}_{k,T+1} = u^{(j)}_{k,T} - \Delta \cdot (\alpha_k u^{(j)}_{k,T} + v_{i,k})$ for $j = 1, 2,\ k = 1, \ldots, d$.
6. If stopping criterion satisfied exit, otherwise goto 2.

- Discretization.

  Replace continuous basis with discrete basis and associated fast discrete transform.

  The functions T and I are defined on the lattice L.

  If $x + U(x)$ is not on lattice L set $T(x + U(x))$ as linear interpolation of T at the four nearest lattice points around $x + U(x)$.

  Derivatives of T - finite differences, calculated off-line.

  Intensive part of computation - Ψ, Ψ^{-1}.

- Time step.

  Time step - inverse of a bound on trace of Hessian matrix of J.

$$\Delta = \left(d \cdot M \int |\nabla T|^2 dx + \sum_k \alpha_k \right)^{-1}.$$

  M - bound on $|\psi_k|$.

- Smoothing. Often useful to smooth and subsample the template and the data.

III.2.2 Coarse to fine

- Useful to gradually increase the number of coefficients being updated.

- Choose $d_1 < d_2 < \ldots < d_m = d$. Apply algorithm 1 but in item (5) update only $k = 1, \ldots, d_1$. When convergence is observed, update in (5) up to d_2 etc.

III.3 Lattice parameterization

- Penalty - differential operator.

  Relaxation procedure - finite difference approximation to parabolic differential equation.

- Problematic for obtaining large coarse deformations and slow.

  Not recommended for starting. It is however possible to use it for fine detail matching after the spectral approach has converged. This is implemented in Miller et al. (1993).

- Parabolic differential equation can be interpreted as a continuum mechanics type equation.

  Penalty -internal stress forces,

  Data - external force.

- More sophisticated continuum mechanics models for the penalty and data increase computation substantially. See Christensen et al. (1996).

III.3.1 Coarse to fine through blocking - video sequence compression.

- Approach used in motion compensation for video sequence compression. Chalidabhongse & Kuo (1997).

- Divide lattice into large disjoint blocks.

  One displacement vector is found independently for each block: gradient procedure or exhaustive search in constrained region for lowest sum of squares or highest correlation.

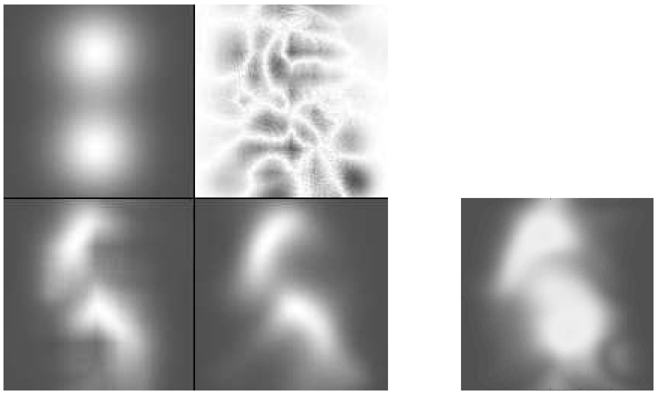

Figure III.3: Right: Upper left hand panel: The template T. Lower right hand panel the data image G. Lower left hand the deformed template after running the non-linear minimization algorithm. Upper right hand panel the difference between the deformed template and the data. Left: The deformed template obtained with the linearized model.

- Smoothly interpolate resulting displacement field. Use this field as initial point for search for displacements of smaller sub-blocks.

  Smoothing not incorporated in a penalty. Results from the smooth interpolation.

- Method very similar to spectral approach if the Haar wavelet basis is used.

III.3.2 Linearization

- U small - linearize: $T(x + U(x)) = T(x) + \nabla T(x) \cdot U(x)$.

  Data term:

  $$\mathcal{D}(U) = \int |T(x) + (\nabla T \cdot U)(x) - I(x)|^2 dx. \tag{III.8}$$

- Solving for d coefficients of U now reduces to simple least squares problem. $u^{(1)}, u^{(2)}$ solve

  $$\sum_{l=1}^{d} \sum_{j=1}^{2} u_l^{(j)} \int T_j \psi_l \psi_k T_i dx = \int (I - T) T_i \psi_k dx, \tag{III.9}$$

for $k = 1, \ldots, d, i = 1, 2$.

Solve with standard linear techniques.

- Results not as good as non-linear approach. See figure III.3.

- Lattice parameterization with differential penalty as in equation III.4, $\mathcal{L} = \nabla$ Euler equations for the linearized cost function same as optical flow and image sequence analysis equations, Horn & Schnuck (1981), Huang & Tsai (1981),Nagel (1983),Terzopoulos (1986).

$$\alpha(\Delta + \epsilon)U^{(1)} + (\nabla T \cdot U)\frac{\partial T}{\partial x_1} = (I(x) - T(x))\frac{\partial T}{\partial x_1}$$
$$\alpha(\Delta + \epsilon)U^{(2)} + (\nabla T \cdot U)\frac{\partial T}{\partial x_2} = (I(x) - T(x))\frac{\partial T}{\partial x_2}.$$

The difference between I and T is replaced by the time derivative of the image sequence.

III.4 Discussion

- Photometric invariance only through prenormalization of gray levels.

- Geometric invariance achieved by modeling deformations of template: $T \circ \theta^{-1}$.

- Initialization - crucial due to the non-linear nature of cost function.

- Important to adjust parameters to the specific problem at hand.

  Choice of orthogonal basis, relative weight of smoothing penalty and the data term, the weights in the smoothing term. No well formulated procedure for making these decisions. In specific contexts such as for example matching of images of a specific anatomy, some experimentation will lead to a reasonable set of parameters.

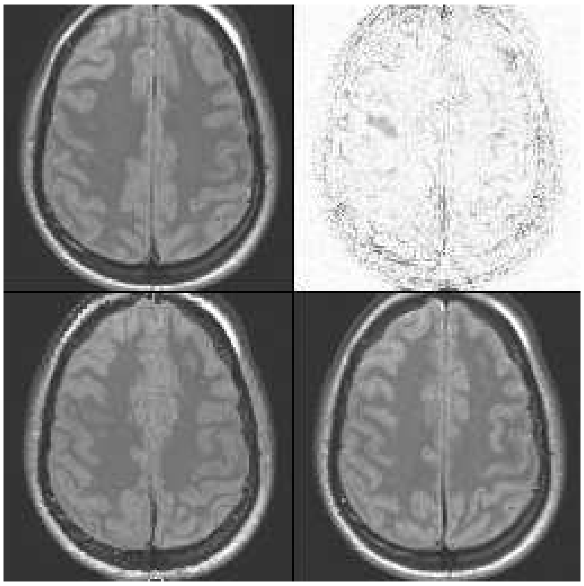

Figure III.4: Example of a warping of one axial MRI brain image to another. Top left: Template T. Bottom left: Deformed template $T(\phi)$, Bottom right, the data image I. Top Right: The error: $|T(\phi) - I|$ - bright is low error value.

- Choosing a 'template image' T is problematic. In examples above a random image from the collection is chosen. This image may be 'far' from most others. Unclear how to estimate some 'mean' template for this type of matching procedure.

- Applications to brain matching

  Elastic matching algorithms in brain research: Friston et al. (1995), Christensen et al. (1996), Sandor & Leahy (1995).

  For 3d data straight forward extension of Algorithm 1.

  The question is whether this fine detail matching has any anatomic or functional meaning.

- A different data model - Single Photon Emission Tomography: Complicated linear transformation of image with Poisson noise. See Amit & Manbeck (1993).

- The connection to the elastic 1-d algorithm:

  Inside - Outside model corresponds to 2d model with template $T = \mu_{in}$ inside the initial curve and μ_{out} outside.

  Difference is in the way the deformations of the initial curve are parameterized.

  Contour is deforming together with the entire 2-d unit square.

  If $\phi(x) = x + U(x)$ is the 2-d deformation then $\gamma(t, \phi) = \phi^{-1}(\gamma_0(t))$.

  1 - d model: an elastic rubber band.

  2 - d model: a curve drawn on an elastic sheet.

  In the second case smaller chance of getting self intersections.