15.5 Conclusions

The posterior Cramér-Rao bound is a convenient tool for evaluating sequential algorithms. A general recursive expression with limited computational and memory requirements has been presented in this work. The bound gives a lower limit on the one step ahead mean square prediction error, and bounds for filtering and smoothing follow similarly. The posterior Cramér-Rao bound yields a natural approach to assess the relative performance in a Monte Carlo evaluation. This has been illustrated using three different simulation-based algorithms and one numerical integration algorithm applied to the terrain navigation problem. The terrain navigation application is well suited to a Bayesian approach in general, and for sequential Monte Carlo methods in particular. All four nonlinear filters yield close to the optimal performance predicted by the Cramér-Rao bound in the simulation evaluation conducted over a commercial terrain map.

16

Statistical Models of Visual Shape and Motion

Andrew Blake
Michael Isard
John MacCormick

16.1 Introduction

This paper[1] addresses some problems in the interpretation of visually observed shapes, both planar and three-dimensional, in motion. Mumford (1996), interpreting the Pattern Theory developed over a number of years by Grenander (1976), views images as pure patterns that have been distorted by a combination of four kinds of degradations. This view applies naturally to the analysis of static, two-dimensional images. The four degradations are given here, together with comments on how they need to be extended to take account of three-dimensional objects in motion.

(i) *Domain warping*, in which the domain of an image I is transformed by a mapping g:

$$I(\mathbf{r}) \rightarrow I(g(\mathbf{r})).$$

The three-dimensional nature of the world means that the warp g may be composed largely of projective or affine transformations. The dynamic nature of the problems addressed here will require time-varying warps $g(\mathbf{r}, t)$.

(ii) *Superposition:* objects may overlap, and in certain forms of imaging this may produce linear combinations. This is fortunate because such combinations can be analysed by linear spectral decomposition. In images of opaque.three dimensional objects. however, far surfaces are obscured by near ones.

[1] This is a modified version of (Blake. Bascle. Isard and MacCormick 1998). and we are grateful to the Royal Society for permission to reprint this material.

(iii) *Distortion and noise:* image measurements are corrupted by noise and blur:

$$I(\mathbf{r}) \rightarrow f(I(\mathbf{r}), \mathbf{n}).$$

Image degradations may be most effectively modelled as applying to certain image features obtained by suitable pre-processing of an image, rather than directly to an image itself.

(iv) *Observation failure:* disturbance of the observation process: often caused, in the work described here, by distracting background clutter.

A key idea in pattern theory is recognition by synthesis, in which predictions following from particular hypotheses play an important role. The predictions are generated and tested against the products of analysis of an image. Bayesian frameworks, which have gained significant influence in modelling perception (Knill, Kersten and Yuille 1996), seem to be a natural vehicle for this combination of analysis and synthesis. In the context of machine perception of shapes, we can state the problem as one of interpreting a *posterior* density function $p(\mathbf{X}|\mathbf{Z})$ for a shape $\mathbf{X}$ in some appropriate *shape-space* $\mathcal{S}$, given data $\mathbf{Z}$ from an image (or data $(\mathbf{Z}_1, \mathbf{Z}_2, \dots)$ from a sequence of images). The posterior density must be computed in terms of *prior* knowledge about $\mathbf{X}$ and inference about $\mathbf{X}$ based on the *observations* $\mathbf{Z}$. Bayes' formula expresses this as

$$p(\mathbf{X}|\mathbf{Z}) \propto p(\mathbf{Z}|\mathbf{X})p_0(\mathbf{X}), \tag{16.1.1}$$

in which $p_0(\mathbf{X})$ is the prior density for $\mathbf{X}$ and the conditional density $p(\mathbf{Z}|\mathbf{X})$ conveys the range of likely observations to arise from a given shape $\mathbf{X}$. All this connects directly to the four degradations above. In particular, type 1 (warping) is represented in the prior p_0. Types 3 and 4 (noise and observation failure) are incorporated into the observation density $p(\mathbf{Z}|\mathbf{X})$.

The framework for Bayesian inference of visual shape and motion that forms the basis of this paper is set out in detail in (Blake and Isard 1998). Here, we aim to summarise that framework and introduce several new ideas. The organisation of this chapter is summarised by section, as follows.

16.2. *Statistical modelling of shape* — how to choose a suitable shape-space $\mathcal{S}$ and a prior p_0, or to learn them from a set of examples.

16.3. *Statistical modelling of image observations* — how to construct an effective observation density $p(\mathbf{Z}|\mathbf{X})$ that takes into account image intensities, both within the shape of interest and in the background.

16.4. *Sampling methods* – using random sample generation to construct an approximate representation of the posterior for $\mathbf{X}$, given that the complexity of $p(\mathbf{Z}|\mathbf{X})$ can render exact representation of the posterior infeasible.

16.5. *Modelling dynamics* – extending the Bayesian framework to deal with sequences of images demands priors for temporal sequences $\mathbf{X}_1, \mathbf{X}_2, \dots$. These can either be constructed by hand or learned from examples.

16.6. *Learning dynamics* – the most effective way to set up dynamic models is to learn them from training sets.

16.7. *Particle filtering* — applied to the interpretation of shapes in motion.

16.8. *Dynamics with discrete states* — extending the dynamical repertoire to modelling of motion with several modes, for example walk–trot–canter–gallop.

16.2 Statistical modelling of shape

This section addresses the construction of a prior model $p_0(\mathbf{X})$ for a shape. This can be done in a somewhat general way if the dimensionality of the shape-space $\mathcal{S}$ is fixed in advance to be small, for example just translations in the plane. Then extended observation of the positions of moving objects in some area can be summarised as a histogram which serves as an approximate representation of the prior p_0 (Fernyhough, Cohn and Hogg 1996). In higher dimensional shape-spaces, involving three-dimensional rigid motion and deformation of shape, histograms are less practical. Here we focus on Gaussian distributions.

A Gaussian distribution is specified by its mean and covariance, and these can be estimated from a training sequence $\mathbf{X}_1, \mathbf{X}_2, \dots$ of shapes by taking the sample mean $\overline{\mathbf{X}}$ and the sample covariance

$$\Sigma = \frac{1}{M} \sum_{k=1}^{M} (\mathbf{X}_k - \overline{\mathbf{X}})(\mathbf{X}_k - \overline{\mathbf{X}})^T.$$

Moreover, Principal Components Analysis (PCA) (Rao 1973) can be used to restrict the shape-space $\mathcal{S}$ to explain most of the variance in the training set while keeping the dimension of $\mathcal{S}$ small, in the interests of computational efficiency (Cootes, Taylor, Lanitis, Cooper and Graham 1993, Baumberg and Hogg 1994, Lanitis, Taylor and Cootes 1995, Beymer and Poggio 1995, Baumberg and Hogg 1995a, Vetter and Poggio 1996, Bascle and Blake 1998). An example is given in figure 16.1.

However, the resulting shape-space, though economical, is not especially easy to interpret because Principal Components need not be meaningful. More meaningful constructive shape-spaces can be generated by acknowledging three-dimensional projective effects and constructing affine spaces, for instance, whose components are directly related to rigid body transformations (Ullman and Basri 1991, Koenderink and van Doorn 1991). In

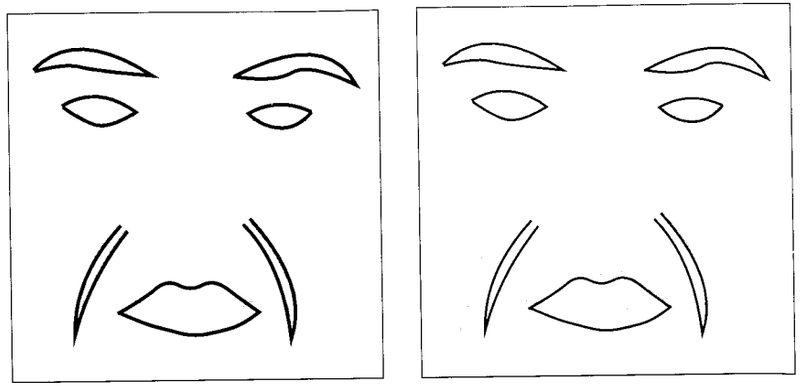
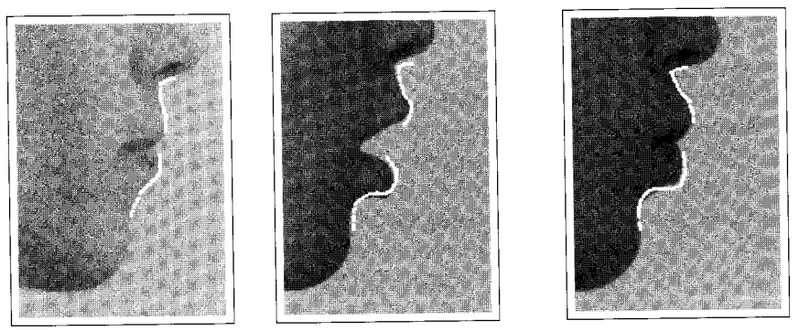


Figure 16.1. **PCA for faces.** *A shape-space of facial expressions is reduced here by PCA to the two-dimensional space that best covers the expressions in a certain training sequence.*

addition, named deformations can be included in a basis for $\mathcal{S}$ as key-frames (Blake and Isard 1994), as in figure 16.2.



Template $\mathbf{Q}_0$ *Key-frame: opening* $\mathbf{Q}_1$ *Key-frame, protrusion* $\mathbf{Q}_2$

Figure 16.2. **Key-frames.** *Lips template followed by two key-frames, representing interactively tracked lips in characteristic positions. The key-frames are combined linearly with appropriate rigid degrees of freedom, to give a shape-space suitable for use in a tracker for non-rigid motion.*

A constructive shape-space S^c can be combined with PCA to give the best of both worlds. 'Residual PCA operates on a constructive shape-space that does not totally cover a certain dataset, supplying missing components by PCA. Then the constructive subspace retains its interpretation and only the residual components, covered by PCA, cannot be directly interpreted. This is done by constructing a projection operator E^c that maps $\mathcal{S}$ to $\mathcal{S}^c$

and applying PCA to the residual training-set vectors $\mathbf{X}_1^r, \mathbf{X}_2^r, \ldots$ where

$$\mathbf{X}^r = \mathbf{X} - E^c\mathbf{X}.$$

Full details of the algorithm are given in (Blake and Isard 1998) and an example of its application is shown in figure 16.3.

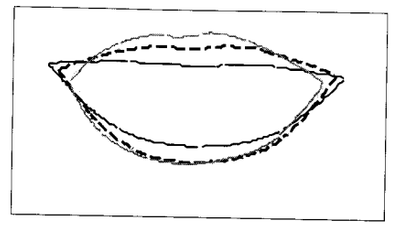


Figure 16.3. **Sampling from a prior for lip-shape, excluding translation** *Random sampling illustrates how a learned prior represents plausible lip configurations. Any rigid translations in the training set, due to head-motion, are separated out as a constructive shape-space in residual PCA.*

16.3 Statistical modelling of image observations

Gaussian distributions may often be acceptable as models of prior shape, but they are adequate as observation distributions only in the clutter-free case. Typically, in our framework, observations are made along a series of splines, normal to the hypothesised shape, as in figure 16.4. Consider the one-dimensional problem of observing, along a single spline, the set of feature positions $\{\mathbf{z} = (z_1, z_2, \ldots, z_M)\}$. Assuming a uniform distribution of background clutter, and a Gaussian model for error in measurement of the position of the true object edge, leads (Isard and Blake 1996) to the multi-modal observation density $p(\mathbf{z}|\mathbf{X})$ depicted in figure 16.5. The multiple peaks in the density are generated by clutter and cannot possibly be modelled by a single Gaussian. A mixture of Gaussians might be feasible, but a very efficient alternative is to use random sampling (next section).

The observation model above was based on the assumption that the observable contour is a "bent wire" resting on a cluttered background. This is not very realistic. It is highly desirable in practice to allow for object opacity and to distinguish between the textured interior of an object and its cluttered exterior. A probabilistic model that reflects this is based on the following assumptions.

Feature localisation error. It is assumed that the feature detector reports object outline position with an error whose density is $\mathcal{E}(\cdot)$, taken usually to be Gaussian.

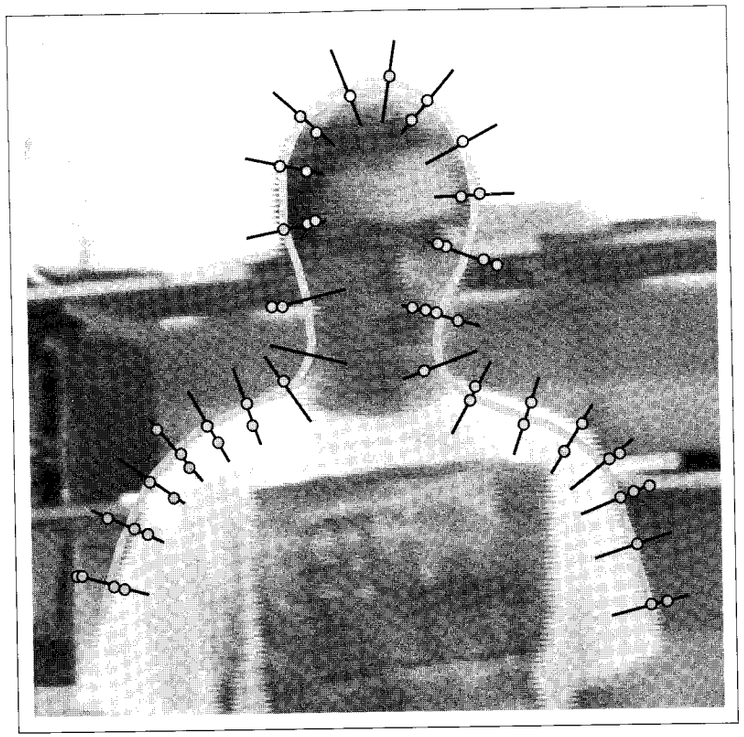


Figure 16.4. **Observation process.** *The thick line is a hypothesised shape, represented as a parametric spline curve. The spines are curve normals along which high-contrast features (white circles) are sought.*

Occlusion probability. The possibility that the outline is missed by the feature detector is allowed, with probability q.

Clutter model. Detection of clutter features is regarded as an i.i.d. random process on the portion of each measurement line that lies outside the object. The probability $\pi(n)$ that n clutter features are detected on a normal is generally taken to be uniform.

Interior model. Interior features on a measurement line are modelled as uniformly distributed along the interior portion of the normal. The distribution $\rho(m)$ for the number m of interior features observed is taken to be Poisson with a known density parameter, which is actually learned by observing instances of the object.

A density $p(\mathbf{z}|\mathbf{X})$ based on these assumptions can be constructed and expressed as $p = \lambda D$ where λ is a constant and

$$D(\mathbf{X}) = p(\mathbf{z}|\mathbf{X})/p(\mathbf{z}|\text{no object present})$$

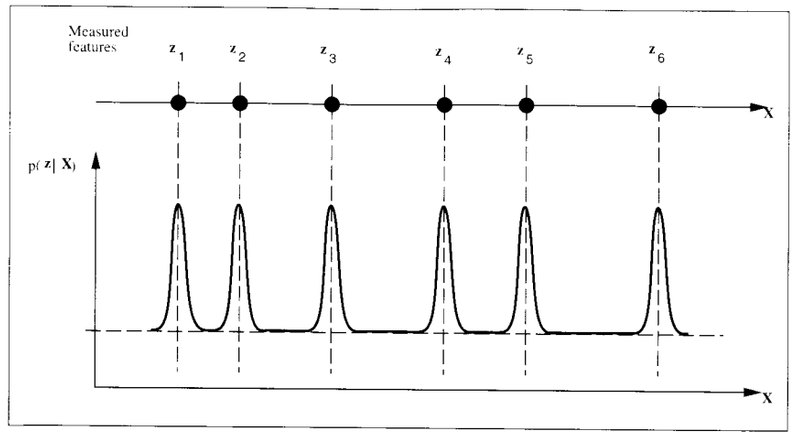


Figure 16.5. **Multi-modal observation density** *(one-dimensional illustration). A probabilistic observation model allowing for clutter and the possibility of missing the target altogether is specified here as a conditional density $p(\mathbf{z}|\mathbf{X})$. It has a peak corresponding to each observed feature.*

— the *contour discriminant*. This is a discriminant function (Duda and Hart 1973) in the form of a *likelihood ratio*. It has the attraction that, in addition to conveying the relative values of $p(\mathbf{z}|\mathbf{X})$, its absolute value is meaningful: $D(\mathbf{X}) > 1$ implies that the observed features $\mathbf{z}$ are more likely to have arisen from the object in location $\mathbf{X}$ than from clutter.

Lastly, densities $p(\mathbf{z}|\mathbf{X})$ for each normal need to be combined into a grand observation density $p(\mathbf{Z}|\mathbf{X})$, and this raises some issues about independence of measurements along an object contour. Details of the form and computation of the full observation density are given in (MacCormick and Blake 1998). Results of the evaluation of the contour discriminant on a real image are shown in figure 16.6. Analysis of the same image using the simpler bent wire observation model degrades the results, failing altogether to locate the leftmost of the three people. The explicit modelling of object opacity has clearly brought significant benefits.

16.4 Sampling methods

The next stage of the pattern recognition problem is to construct the posterior density $p(\mathbf{X}|\mathbf{Z})$ by applying Bayes' rule (16.1.1). In the previous section, it became plain that the observation density has a complex form in clutter. This means that direct evaluation of $p(\mathbf{X}|\mathbf{Z})$ is infeasible. However, iterative sampling techniques can be used (Geman and Geman 1984, Ripley and Sutherland 1990, Grenander, Chow and Keenan 1991, Storvik 1994). The *factored sampling* algorithm (Grenander et al. 1991) — also known

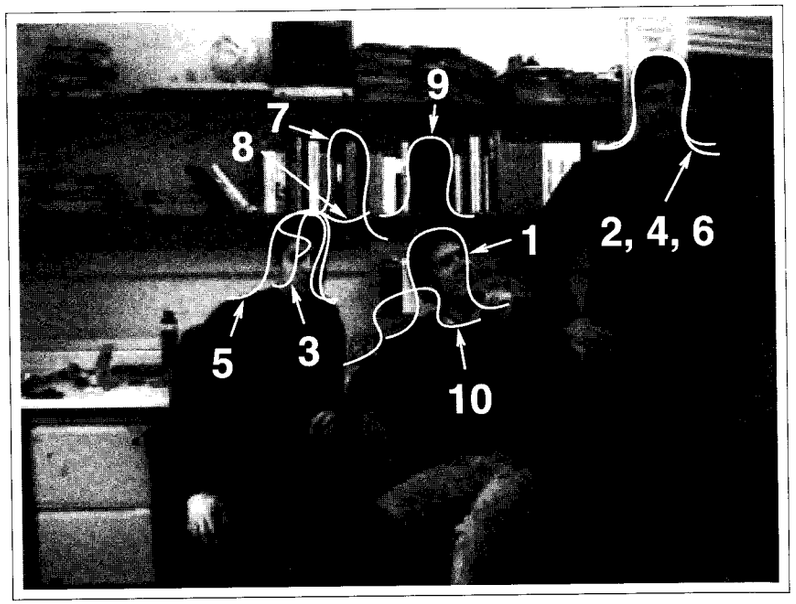


Figure 16.6. **Finding head-and-shoulders outlines in an office scene.** *The results of a sample of* 1,000 *configurations are shown ranked by value of their contour discriminant. The figure displays the cases in which* $D > 1$*, indicating a configuration that is more target-like than clutter-like.*

as sampling importance resampling, or SIR (Rubin 1988) — generates a random variate $\mathbf{X}$ from a distribution $\tilde{p}(\mathbf{X})$ that approximates the posterior $p(\mathbf{X}|\mathbf{Z})$. First a sample-set $\{\mathbf{s}^{(1)}, \ldots, \mathbf{s}^{(N)}\}$ is generated from the prior density $p(\mathbf{x})$ and then a sample $\mathbf{X} = \mathbf{X}_i$, $i \in \{1, \ldots, N\}$ is chosen with probability

$$\pi_i = \frac{p(\mathbf{Z}|\mathbf{X} = \mathbf{s}^{(i)})}{\sum_{j=1}^{N} p(\mathbf{Z}|\mathbf{X} = \mathbf{s}^{(j)})}.$$

Sampling methods have proved remarkably effective for recovering static objects from cluttered images. For such problems, $\mathbf{X}$ is a multi-dimensional set of parameters for curve position and shape. In that case the sample-set $\{\mathbf{s}^{(1)}, \ldots, \mathbf{s}^{(N)}\}$ is drawn from the posterior distribution of $\mathbf{X}$-values, as illustrated in figure 16.7.

16.5 Modelling dynamics

In order to be able to interpret moving shapes in sequences of images, it is necessary to supply a prior distribution, not just for shape, but also for

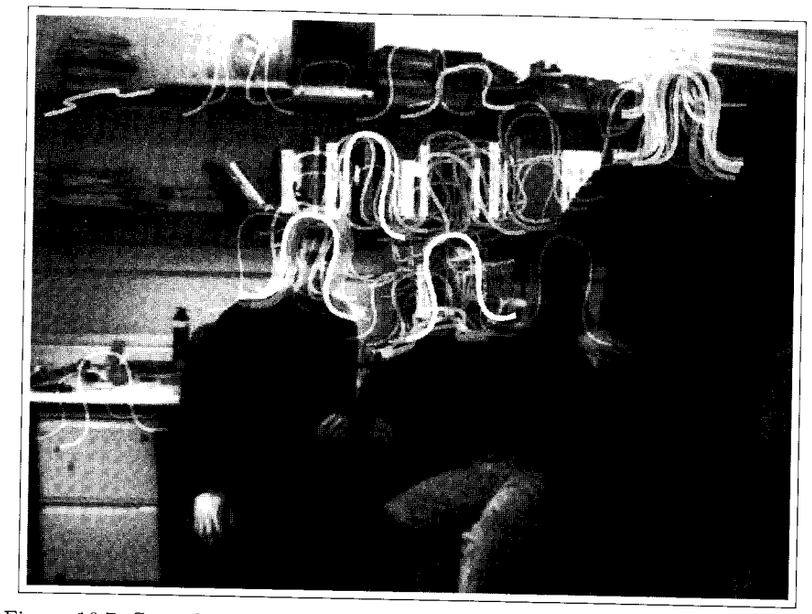


Figure 16.7. **Sample-set representation of posterior shape distribution** *for a curve with parameters* $\mathbf{X}$*, modelling a head outline. Each sample* $\mathbf{s}^{(n)}$ *is shown as a curve (of varying position and shape) with a mass proportional to the weight* $\pi^{(n)}$*. The prior is uniform over translation, with some constrained Gaussian variability in the remainder of its affine shape-space.*

the motion of that shape. Consider the problem of building an appropriate prior model for the position of a hand-mouse engaged in an interactive graphics task. A typical trace in the x-y plane of a finger drawing letters is given in figure 16.8. If the entire trajectory were treated as a training set, the methods discussed earlier could be applied to learn a Gaussian prior distribution for finger position. The learned prior is broad, spanning a sizeable portion of the image area, and places little constraint on the measured position at any given instant. Nonetheless, it is quite clear from the figure that successive positions are tightly constrained. Although the prior covariance ellipse spans about 300×50 pixels, successive sampled positions are seldom more than 5 pixels apart.

For sequences of images, then, a global prior $p_0(\mathbf{X})$ is not enough. What is needed is a conditional distribution $p(\mathbf{X}_k|\mathbf{X}_{k-1})$ giving the distributions of possibilities for the shape $\mathbf{X}_k$ at time $t = k\tau$ *given* the shape $\mathbf{X}_{k-1}$ at time $t = (k-1)\tau$ (where τ is the time-interval between successive images). This amounts to a first order Markov chain model in shape space in which, although in principle $\mathbf{X}_k$ may be correlated with all of $\mathbf{X}_1 \ldots \mathbf{X}_{k-1}$, only correlation with the immediate predecessor is explicitly acknowledged.

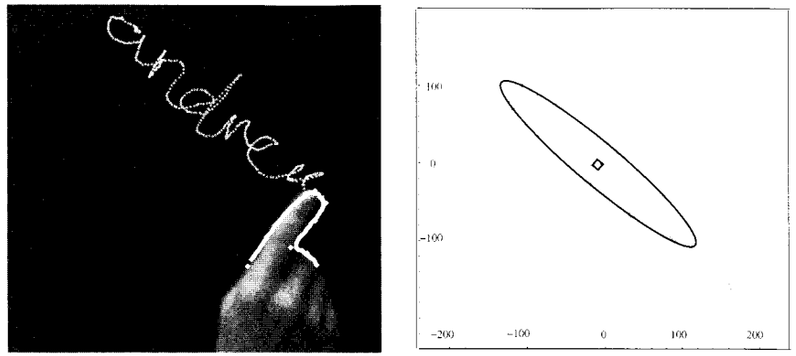


Figure 16.8. **The moving finger writes.** *The finger trajectory (left), which has a duration of about 10 seconds, executes a broad sweep over the plane. When the trajectory is treated as a training set, the learned Gaussian prior is broad, as the covariance ellipse (right) shows. Clearly, though, successive positions (individual dots represent samples captured every 20ms) are much more tightly constrained.*

For the sake of tractability, it is reasonable to restrict Markov modelling to linear processes. In principle and in practice it turns out that a first order Markov chain is not quite enough, generally, but second order suffices. The detailed arguments for this, addressing such issues as capacity to represent oscillatory signals and trajectories of inertial bodies, can be found in (Blake and Isard 1998, Chapter 9). Figure 16.9 illustrates the point for a practical example. A second order, Auto-Regressive Process (ARP) is most concisely expressed by defining a state vector

$$\mathcal{X}_k = \begin{pmatrix} \mathbf{X}_{k-1} \\ \mathbf{X}_k \end{pmatrix}, \tag{16.5.2}$$

and then specifying the conditional probability density $p(\mathcal{X}_k|\mathcal{X}_{k-1})$. In the case of a linear model, this can be done constructively as follows:

$$\mathcal{X}_k - \overline{\mathcal{X}} = A(\mathcal{X}_{k-1} - \overline{\mathcal{X}}) + B\mathbf{w}_k, \tag{16.5.3}$$

where

$$A = \begin{pmatrix} 0 & I \\ A_2 & A_1 \end{pmatrix}, \quad \overline{\mathcal{X}} = \begin{pmatrix} \overline{\mathbf{X}} \\ \overline{\mathbf{X}} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 0 \\ B_0 \end{pmatrix}. \tag{16.5.4}$$

Each $\mathbf{w}_k$ is a vector of N_X independent random $\mathcal{N}(0,1)$ variables and $\mathbf{w}_k$, $\mathbf{w}_{k'}$ are independent for $k \neq k'$. This specifies the probable temporal evolution of the shape $\mathbf{X}$ in terms of parameters A, B, and covers multiple oscillatory modes and/or constant velocity motion. The constructive form is attractive because it is amenable to direct simulation, simply by supplying a realisation of the succession of random variates $\mathbf{w}_k$.

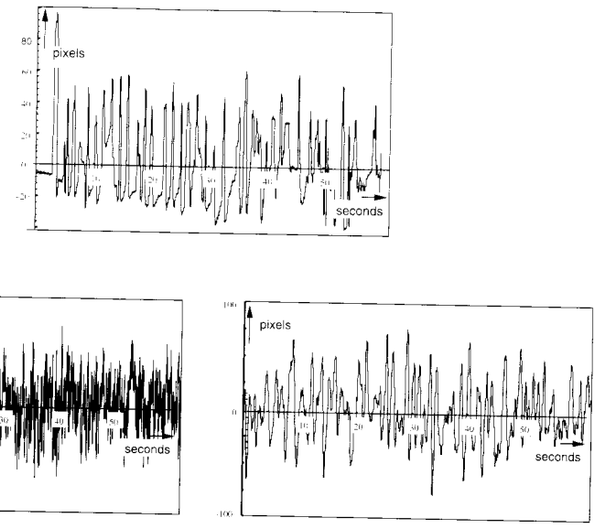


Figure 16.9. **Motion data from talking lips.** *Training sequence of 60 seconds duration (top). Random simulations of learned models — 1st order (left) and 2nd order (right). Only the 2nd order model captures the natural periodicity (around 1 Hz) of the training set, and spectrogram analysis confirms this.*

16.6 Learning dynamics

Motion parameters (A, B in this paper) can be set by hand to obtain desired effects, and a logical approach to this has been developed (Blake and Isard 1998, chapter 9). Experimentation allows these parameters to be refined by hand for improved tracking, but this is a difficult and unsystematic business. It is far more attractive to learn dynamic models on the basis of training sets. A number of alternative approaches have been proposed for learning dynamics, with a view to gesture-recognition -- see for instance (Mardia, Ghali, Howes, Hainsworth and Sheehy 1993, Campbell and Bobick 1995, Bobick and Wilson 1995). The requirement there is to learn models that are sufficiently finely tuned to discriminate among similar motions. In our context of the problem of motion tracking, rather different methods are required so as to learn models that are sufficiently coarse to encompass all likely motions.

Initially, a hand-built model is used in a tracker to follow a training sequence that must be not be too hard to track. This can be achieved by allowing only motions that are not too fast and limiting background clutter, or eliminating it using background subtraction (Baumberg and Hogg 1994, Murray and Basu 1994, Koller, Weber and Malik 1994, Rowe and Blake 1996). Once a new dynamic model has been learned, it can be used to build a more competent tracker, one that is specifically tuned to

the sort of motions it is expected to encounter. This can be used either to track the original training sequence more accurately, or to track a new and more demanding training sequence involving greater agility of motion. The cycle of learning and tracking is illustrated in figure 16.10. Typically two

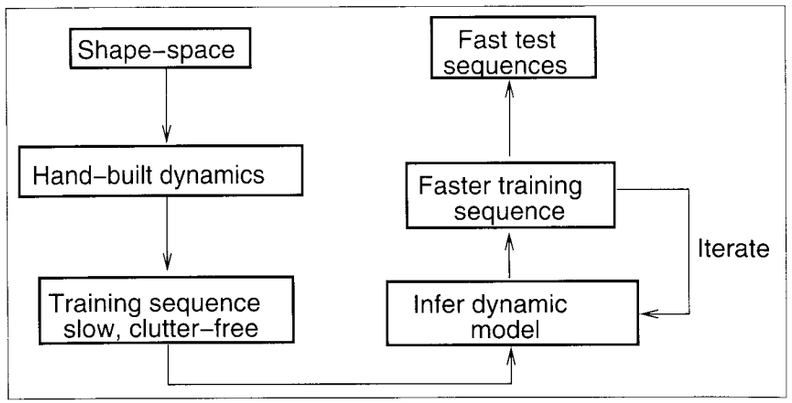


Figure 16.10. **Iterative learning of dynamics.** *The model acquired in one cycle of learning is installed in a tracker to interpret the training sequence for the next cycle. The process is initialised with a tracker whose prior is based on a hand-built dynamic model.*

or three cycles suffice to learn an effective dynamic model.

In mathematical terms, the general problem is to estimate the coefficients A_1, A_2, $\overline{\mathbf{X}}$ and B from a training sequence of shapes $\mathbf{X}_1, \ldots, \mathbf{X}_M$, gathered at the image sampling frequency. Known algorithms to do this are based on the Maximum Likelihood principle (Rao 1973, Kendall and Stuart 1979) and use variants of Yule-Walker equations for estimation of the parameters of auto-regressive models (Gelb 1974, Goodwin and Sin 1984, Ljung 1987). Suitable adaptations for multidimensional shape-spaces are given by (Blake and Isard 1994, Baumberg and Hogg 1995b, Blake, Isard and Reynard 1995), with a number of useful extensions in (Reynard, Wildenberg, Blake and Marchant 1996). One example is the scribble in figure 16.11, learned from the training-sequence in figure 16.8.

A more complex example consists of learning the motions of an actor's face, using the shape-space described earlier that covers both rigid and non-rigid motion. Figure 16.12 illustrates how much more accurately realistic facial motion can be represented by a dynamic model actually learned from examples.

The learning algorithms referred to above treat the training set as exact, whereas in fact it is inferred from noisy observations. Dynamics can be learned directly from the observations using Expectation-Maximisation (EM) (Dempster, Laird and Rubin 1977). Learning dynamics by EM is

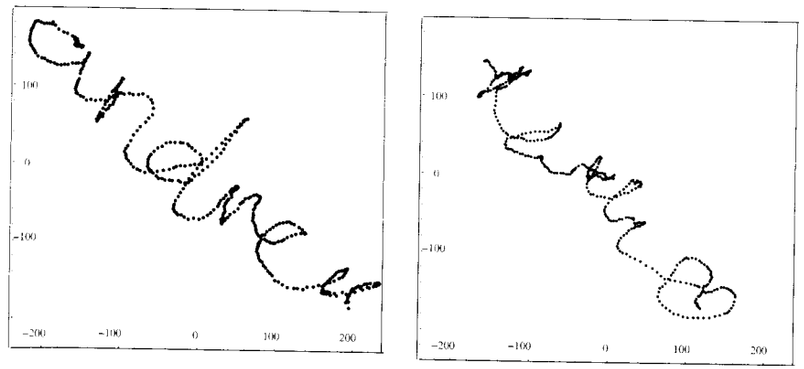


Figure 16.11. **Scribbling: simulating a learned model for finger-writing.** *A training set (left) consisting of six handwritten letters is used to learn a dynamical model for finger motion. A random simulation from the model (right) exhibits reasonable gross characteristics.*

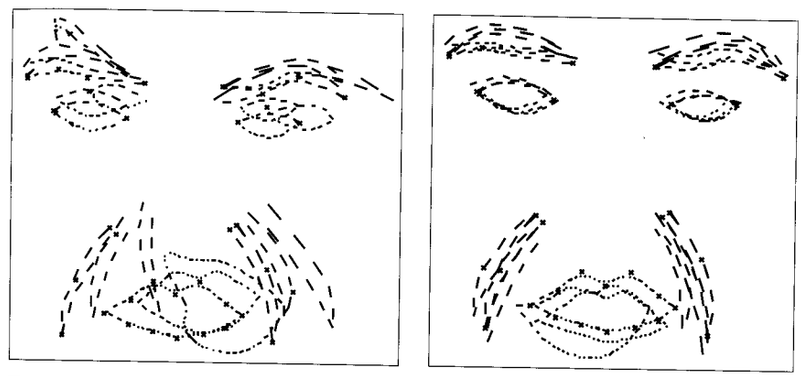


Figure 16.12. **Trained dynamics for facial motion.** *Hand-built dynamics, exhibited here by random simulation (left) are just good enough, when used in tracking, to gather a training sequence. Trained dynamics (right) however, capture more precisely the constraints of realistic facial motion.*

suggested by Ljung (1987) and the detailed algorithm is given in (North and Blake 1997). It is related to the Baum-Welch algorithm used to learn speech models (Huang, Arika and Jack 1990, Rabiner and Bing-Hwang 1993) but with additional complexity because the state-space is continuous rather than discrete. In practice, accuracy of the learned dynamics is significantly improved when EM is used, especially in the case of more coherent oscillations.

An extension of the basic algorithm for *classes* of objects, dealing independently with motion and with variability of mean shape/position over

the class, is described in (Reynard et al. 1996). The same algorithm is also used for modular learning — the aggregation of training sets for which a joint dynamic model is to be constructed.

16.7 Particle filtering

The principles of particle filtering have been developed earlier in the book. Particle filtering has been introduced into the practice of Computer Vision (Isard and Blake 1996, Isard and Blake 1998a) over the past few years, where it is known as the CONDENSATION algorithm, and has been applied to great effect for tracking moving objects in image sequences.

Each time-step of the particle filter generates a weighted, time-stamped sample-set, denoted $\mathbf{s}_k^{(n)}$, $n = 1, \ldots, N$ with weights $\pi_k^{(n)}$, representing approximately the conditional state-density $p(\mathcal{X}_k|\underline{\mathbf{Z}}_k)$ at time $t = k\tau$, where $\underline{\mathbf{Z}}_k = (\mathbf{Z}_1, \ldots, \mathbf{Z}_k)$, the history of observations. How is this sample-set obtained? Clearly, the process must begin with a prior density, and the effective prior for time-step k should be $p(\mathcal{X}_k|\underline{\mathbf{Z}}_{k-1})$. This prior is of course multi-modal in general and no functional representation of it is available. It is derived from the representation as a sample set $\{(\mathbf{s}_{k-1}^{(n)}, \pi_{k-1}^{(n)}),\ n = 1, \ldots, N\}$ of $p(\mathcal{X}_{k-1}|\underline{\mathbf{Z}}_{k-1})$, the output from the previous time-step, to which prediction must then be applied.

The iterative process applied to the sample-sets is depicted in figure 16.13. At the top of the diagram, the output from time-step $k-1$ is the weighted sample-set $\{(\mathbf{s}_{k-1}^{(n)}, \pi_{k-1}^{(n)}),\ n = 1, \ldots, N\}$. The aim is to maintain, at successive time-steps, sample sets of fixed size N, so that the algorithm can be guaranteed to run within a given computational resource. The first operation therefore is to sample (with replacement) N times from the set $\{\mathbf{s}_{k-1}^{(n)}\}$, choosing a given element with probability $\pi_{k-1}^{(n)}$. Some elements, especially those with high weights, may be chosen several times, leading to identical copies of elements in the new set. Others with relatively low weights may not be chosen at all.

Each element chosen from the set is now subjected to a predictive step, using an ARP dynamic model, as in equation (16.5.3). This involves sampling a value of $\mathcal{X}_k$ randomly from the conditional density $p(\mathcal{X}_k|\mathcal{X}_{k-1})$ to form a new set member $\mathbf{s}_k^{(n)}$. Since the predictive step includes a random component, identical elements may now split as each undergoes its own independent random motion step. At this stage, the sample set $\{\mathbf{s}_k^{(n)}\}$ for the new time-step has been generated but, as yet, without its weights: it is approximately a fair random sample from the effective prior density $p(\mathcal{X}_k|\underline{\mathbf{Z}}_{k-1})$ for time-step $t = k\tau$. Finally, the observation step from factored sampling is applied, generating weights from the observation density $p(\mathbf{Z}_k|\mathcal{X}_k)$ to obtain the sample-set representation $\{(\mathbf{s}_k^{(n)}, \pi_k^{(n)})\}$ of state-density for time t.

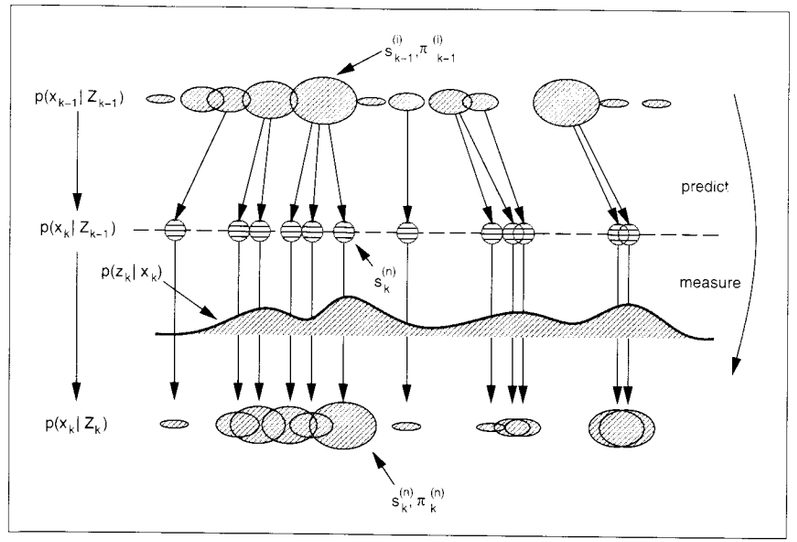


Figure 16.13. **One time-step in the CONDENSATION algorithm.** *Blob centres represent sample values and sizes depict sample weights.*

In general sequential importance sampling, almost any "importance distribution" can be used to propagate particles from one time-step to the next, provided the weights are calculated so as to counteract the effects of the importance distribution. The dynamical model $p(\mathcal{X}_k|\mathcal{X}_{k-1})$ used to propagate members of the sample set here is a particular choice of importance distribution that is effective for the visual tracking problems to be addressed. Of course this distribution cannot guarantee that the sample weights have low variance — indeed, the optimal distribution for this purpose is $p(\mathcal{X}_k|\mathbf{Z}_k, \mathcal{X}_{k-1})$ (Doucet, Godsill and Andrieu 2000), but this distribution is unknown and intractable in the problems considered here. Using $p(\mathcal{X}_k|\mathcal{X}_{k-1})$ as the importance distribution has two advantages: it can be learnt by the methods described in the last section, and samples can be drawn from it easily.

A good deal of experimentation has been conducted in applying the CONDENSATION algorithm to the tracking of visual motion, including moving hands and dancing figures. Perhaps one of the most stringent tests involved the tracking of a leaf on a bush, in which the foreground leaf is effectively camouflaged against the background. Results are shown in figure 16.14 and experimental details can be found in (Isard and Blake 1996).

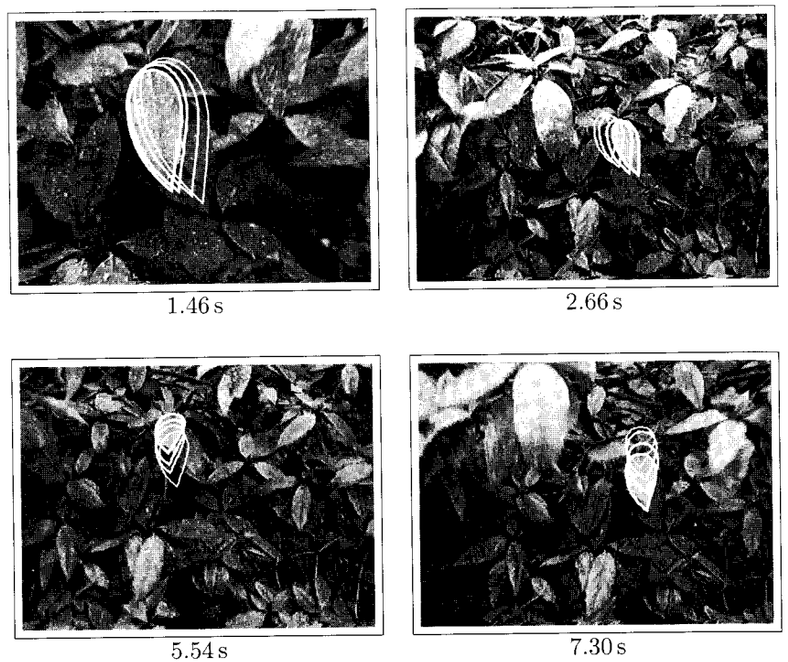


Figure 16.14. **Tracking with camouflage.** *Stills depict mean contour configurations, with preceding tracked leaf positions plotted at 40 ms intervals to indicate motion.*

16.8 Dynamics with discrete states

A recent development of the dynamic models already described is to append to the state variable $\mathcal{X}$ a discrete state y_k to make a mixed state

$$\mathcal{X}_k^+ = \begin{pmatrix} \mathcal{X}_k \\ y_k \end{pmatrix}, \tag{16.8.5}$$

where $y_k \in \{1, \ldots, N_S\}$ is drawn from a finite set of discrete states with integer labels. Each discrete state represents a mode of motion such as stroke, rest and shade for a hand engaged in drawing. Corresponding to each state $y_{k-1} = i$ there is a dynamical model $p_i(\mathcal{X}_k|\mathcal{X}_{k-1})$, which, in the case of the drawing hand, is likely to be an ARP as in (16.5.3). The stroke model, for instance, might represent constant velocity motion, whereas shading would be oscillatory. In addition, and independently, state transitions are governed by

$$P(y_k = j|y_{k-1} = i) = T_{i,j}.$$

a transition matrix following usual practice for Markov chains. More generally, transition probabilities could be made sensitive to the context $\mathcal{X}_{k-1}$ in state-space, so that

$$P(y_k = j|y_{k-1} = i, \mathcal{X}_{k-1}) = T_{i,j}(\mathcal{X}_{k-1}).$$

For example this could be used to express an enhanced probability of transition into the resting state when the hand is moving slowly.

The incorporation of mixed states into the CONDENSATION algorithm is straightforward. It involves using the extended state $\mathcal{X}_k^+$ in place of the original $\mathcal{X}_k$, so that a sample $\mathbf{s}_k^{(n)}$ is now a value of the extended state. The prediction step, which generates a new sample $\mathbf{s}_k^{(n)}$ from an old one $\mathbf{s}_{k-1}^{(n)}$, requires a discrete step and a continuous one. First, the discrete state y_k for the new sample is $y_k = j$, chosen randomly, with probability $T_{i,j}$, where i is the discrete state of the old sample. Then the continuous state is chosen by sampling randomly from a continuous density, as in the original algorithm, but now one of several possible densities $p_i(\mathcal{X}_k|\mathcal{X}_{k-1})$, where again i is the discrete state of the old sample.

Experiments with a three-state model for drawing have been described in detail elsewhere (Isard and Blake 1998b). In addition to enhancing tracking performance, there is the bonus that the current discrete state y_k can be estimated at each time $t = k\tau$, effectively carrying out gesture recognition as a side-effect. One interesting variation on the mixed-state theme uses continuous conditional densities $p_i(\mathcal{X}_k|\mathcal{X}_{k-1})$ which are not ARP models. Consider the example of a moving ball, which may occasionally bounce. This could be represented using two states $\{1, 2\}$ in which $i = 1$ stands for the free ballistic motion of the ball, and $i = 2$ is the bounce event. A suitable transition matrix would be

$$T = \begin{pmatrix} 1-\epsilon & \epsilon \\ 1 & 0 \end{pmatrix},$$

in which $0 < \epsilon \ll 1$ so that ballistic motion has a mean duration τ/ϵ between bounces. The fact that $T_{2,2} = 0$ ensures that the model always returns to ballistic motion after a bounce — bouncing at each of two consecutive time-steps is disallowed. Now $p_1(\ldots|\ldots)$ is an ARP for ballistic motion but $p_2(\ldots|\ldots)$ models the instantaneous reversal of velocity normal to the reflecting surface. Details of experiments with such a model are in (Isard and Blake 1998b), but the results are illustrated in figure 16.15. The use of mixed discrete/continuous states in a particle filter has been proposed independently by (Semerdjiev, Jilkov and Angelove 1998).

16.9 Conclusions

A high-speed tour has been given of a framework for probabilistic modelling of shapes in motion, and of their visual observation. The key points are that

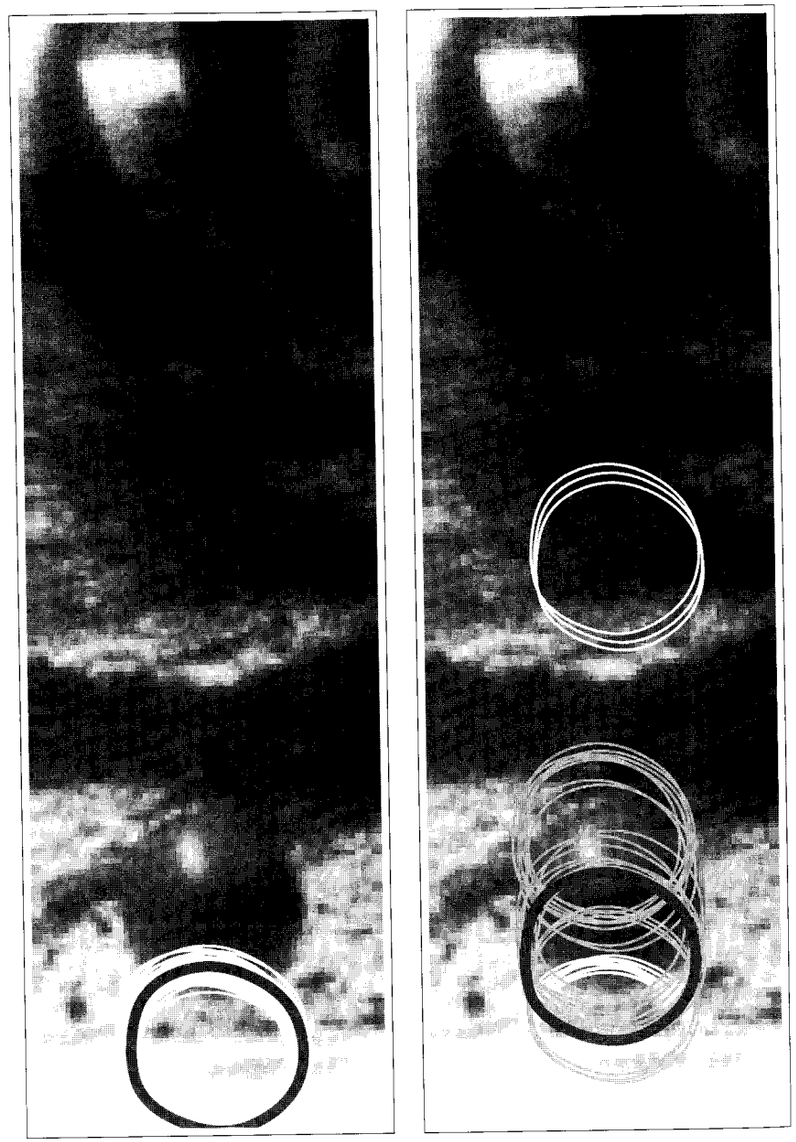


Figure 16.15. **Mixed states tighten constraints in dynamic models.** *A conventional, continuous-state ARP model (left) used to track ballistic motion fails unrecoverably as the ball bounces. Introducing an explicit discrete state for the bounce allows the sample set to split, so that a significant proportion are able to track the bounce.*

visual clutter makes motion analysis difficult, and demands sophisticated probabilistic mechanisms to handle the resulting uncertainty. Further, prior models of motion and observation provide powerful constraints, especially so when the models are learned. A more detailed treatment is given in (Blake and Isard 1998). Since that account, several new modelling tools have been developed. First, the contour discriminant is a new observational model expressed as a likelihood ratio taking opacity of objects into account. Second, complex models for combined rigid and nonrigid motion have been constructed, with a new algorithm for decomposing the two components. Third, extending dynamic states to include discrete labels can significantly enhance their power to constrain perceptual interpretation of shape.

Many interesting questions remain to be addressed. One of these is whether sampling methods for object localisation can be fused elegantly with the CONDENSATION algorithm, to allow robust handling of birth and death (Grenander and Miller 1994) processes in which objects enter and leave the scene. A second is to extend mixed-state models to give reliable gesture recognition on the fly, in a manner that is integrated with the tracking process. A third is to develop algorithms, based on EM, to learn dynamical models from sequences tracked by CONDENSATION, using the full richness of its probabilistic representation, both for continuous and mixed state systems. Finally, another important goal is the extension of particle filtering to allow its use in many dimensions and complex configuration spaces. Such techniques are emerging all the time, and recent contributions include the partitioned sampling of (MacCormick 1999) and the layered sampling of (Sullivan, Blake, Isard and MacCormick 1999).

Acknowledgements

The authors would like to acknowledge the support of the EPSRC.