Learning Dynamics of Complex Motions from Image Sequences

David Reynard[1], Andrew Wildenberg[1], Andrew Blake[1] and John Marchant[2]

[1] Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, UK
[2] Silsoe Research Institute, Bedford, UK.

Abstract. The performance of Active Contours in tracking is highly dependent on the availability of an appropriate model of shape and motion, to use as a predictor. Models can be hand-built, but it is far more effective and less time-consuming to learn them from a training set. Techniques to do this exist both for shape, and for shape and motion jointly. This paper extends the range of shape and motion models in two significant ways. The first is to model jointly the random variations in shape arising within an object-class and those occuring during object motion. The resulting algorithm is applied to tracking of plants captured by a video camera mounted on an agricultural robot. The second addresses the tracking of coupled objects such as head and lips. In both cases, new algorithms are shown to make important contributions to tracking performance.

1 Introduction

The use of Kalman filters [1] to track the motion of objects in real time is now a standard weapon in the arsenal of Computer Vision [14, 11, 8, 10, 16, 3]. A crucial consideration for effective real-time performance is that some form of dynamical model be identified [13, 5, 2] and used as a predictor. In many cases, available models are deterministic — based on ordinary differential equations. However, to be usable in a Kalman filtering framework it is crucial that the model contain both deterministic and stochastic components — stochastic differential equations. Such models can be learned effectively from training data [9, 5].

In this paper we develop two significant elaborations for stochastic dynamical models. The first concerns modelling object classes for objects in motion. The second addresses the efficient modelling of couplings between tracked objects.

1.1 Shape and Motion Variability

The first problem addressed by this paper concerns learning dynamical models which represent both *class* and *dynamical* variability. Class variability arises from the differences between objects in a given class which can be learned effectively by Principal Components Analysis applied to a linear curve parameterisation [6]. With moving objects, dynamical variability must also be considered, modelling changes that occur during motion, due to projective effects and actual physical disturbances.

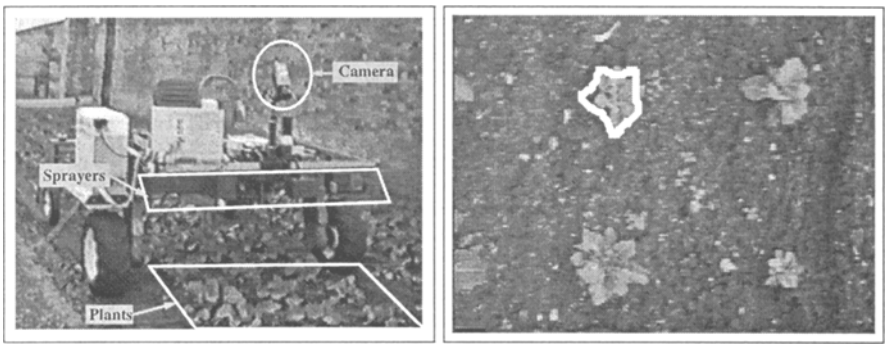

Fig. 1. (a) *An autonomous tractor incorporates a downward pointing camera to monitor plants passing under an array of spray nozzles. Sprayed chemicals can then be directed onto or away from plants, as appropriate.* (b) **Plant tracking.** *A mobile vehicle sees plants (cauliflowers in this instance) passing through the field of view of a downward pointing camera. The plants' motion and shape are captured by a dynamic contour, shown in white.*

Previous systems for learning dynamics have not addressed the important distinction between these two sources of variability [5, 2]. If the two are lumped together into one dynamical model, there will be insufficient constraint on motion variability. A tracker incorporating such a model as a predictor will allow temporal motion/shape changes to range over the *total* modelled variability, both class and temporal. This is inappropriate. Rather, modelled class variability should apply only as a tracker is re-initialised on a new object. Once tracking is underway, the identity of the object does not change, so class variability should be suspended. Instead, the dynamical variability model should take over, relatively tightly constrained as it allows only the variability normally associated with motion. Maintaining the distinction between the two sources of variability is of considerable practical importance.

In a robotic application, a mobile tractor-robot monitoring the motion of plants in a downward pointing camera controls the application of sprayed chemicals from a spray-nozzle array (figure 1 (a)). Moving plants in the video stream are tracked by a dynamic contour tracker (figure 1 (b)). A dynamical model was learnt from video sequences of 36 plants. Inter-class variability proves to be considerably greater than motion variability in this kind of data (figure 2). Modelling them independently ought therefore to enhance temporal stability during tracking. This will indeed prove to be so.

1.2 Coupled Motions

The second problem addressed here is that of learning the dynamics of a coupled pair of systems. The point of attempting to model the coupling is that stability of tracking may be enhanced by the implied constraints. For instance, Moses *et al* [12] built a system to track lip motion and deformation, but it proved difficult, given that lips are long and thin, to stabilise them to horizontal translation of the head. This phenomenon is illustrated in figure 3. Here we show that this

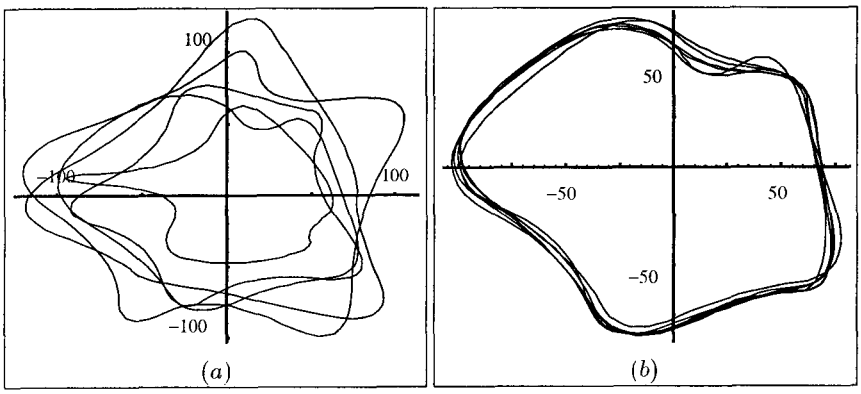

Fig. 2. Class-variability swamps motion-variability. *(a) Measured class-variation over five plants is considerable. In contrast (b) variation due to the motion of one individual plant, as it passes through the camera's field of view, is relatively small.*

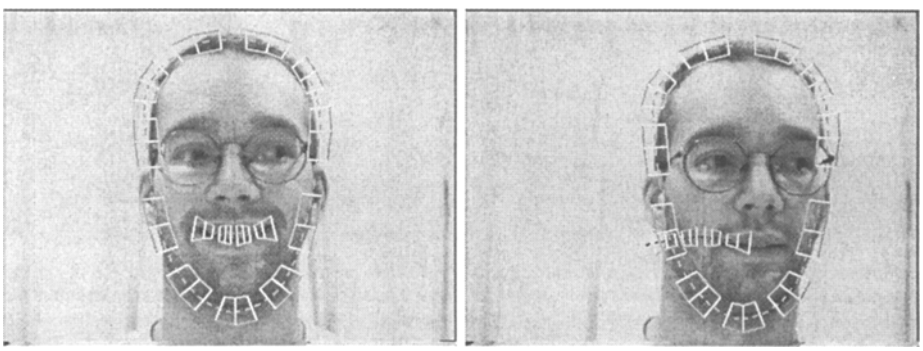

Fig. 3. Uncoupled head and lip tracking is unsuccessful. *The two images show the results of simultaneous head and lip tracking when the dynamics of the two systems are independent. In the first image, the lip tracker follows the movement of the lips well, but as soon as the head begins to translate the position of the lips is lost. This shows that some sort of coupling is necessary.*

defect can be largely removed by using a prior model for motion in which one object, in this case the head, is regarded as primary and the other (the lips) as secondary, driven by the primary.

Modelling this coupling fully and installing it, without approximation, in a Kalman filter tracker — "strong coupling" — incurs a severe increase in computational cost. We therefore propose an approximate "weak coupling" in which state covariance is constrained during tracking (see [16] for a related idea), leading to greatly reduced computational cost. Weak coupling is tested experimentally and found, in practice, to give results that differ remarkably little from the strongly coupled system.

2 Modelling Class and Motion Variability

The distinction between class and motion variability is important. In this section multidimensional, second order dynamical models will be extended to model these two sources of variability independently. The resulting, augmented model, once installed as the predictor for a tracker, performs much more robustly than when all variability is modelled jointly.

The joint dynamical model is a second order stochastic differential equation (SDE) which, in the discrete-time form, is:

$$\mathbf{X}_{n+2} - \overline{\mathbf{X}} = A_1(\mathbf{X}_{n+1} - \overline{\mathbf{X}}) + A_0(\mathbf{X}_n - \overline{\mathbf{X}}) + B\mathbf{w}_n \tag{1}$$

where $\mathbf{X}$ is an N-dimensional state vector parameterising the configuration (position and shape) of the modelled contour. The mean configuration $\overline{\mathbf{X}}$, is a constant in this simple model. Coefficients A_0 and A_1 fix the deterministic components of the model [13]. The stochastic component of the model is a noise source $\mathbf{w}_n$, a vector of N independent unit normal variables, coupled into the system via the $N \times N$ matrix B. An algorithm for learning systems of this kind is known [5, 4] but an improved algorithm is needed to model the two sources of variability independently.

2.1 Extended Dynamical Model

The improved model extends the state vector by regarding the mean configuration $\overline{\mathbf{X}}$ as a variable, rather than a constant as above. Now the dynamical model, most conveniently written in block-matrix form, is:

$$\begin{pmatrix} \mathbf{X}_{n+1} \\ \mathbf{X}_{n+2} \\ \overline{\mathbf{X}}_{n+1} \end{pmatrix} = \begin{pmatrix} 0 & I & 0 \\ A_0 & A_1 & I - A_0 - A_1 \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} \mathbf{X}_n \\ \mathbf{X}_{n+1} \\ \overline{\mathbf{X}}_n \end{pmatrix} + \begin{pmatrix} 0 \\ B\mathbf{w}_n \\ 0 \end{pmatrix} \tag{2}$$

This simply augments the original model (1) with an additional equation stating $\overline{\mathbf{X}}$ is constant over time. It might appear that this is identical to the old model. The crucial difference is that $\overline{\mathbf{X}}$ is no longer known *a priori*, but is estimated online. Once this dynamical system is installed as the predictor for a Kalman filter tracker, $\overline{\mathbf{X}}$ is initialised with its *estimated* mean value $\overline{\mathbf{X}}_0$ and associated variances and covariances which are obtained by statistical estimation from training data. The value of $\overline{\mathbf{X}}_n$ converges rapidly in the Kalman filter and remains fixed, reinitialised only when a new object is to be tracked.

2.2 Learning Algorithm

One of the main theoretical results of this paper is the learning algorithm for the extended system dynamics. Given objects labelled $\gamma = 1, \ldots, \Gamma$, each observed in motion for N_γ timesteps as data sequences $\mathbf{X}_n^\gamma$, the problem is to estimate

global parameters for the dynamics A_0, A_1, B and mean configurations for each object $\overline{\mathbf{X}}^\gamma$ by jointly maximising log likelihood over all the parameters:

$$L(\{\mathbf{X}_n^\gamma\}|A_0, A_1, B, \overline{\mathbf{X}}^1, \ldots, \overline{\mathbf{X}}^\Gamma) = -\frac{1}{2}\sum_{\gamma=1}^{\Gamma}\sum_{n=1}^{N_\gamma-2}\left|B^{-1}R_n^\gamma\right|^2 - \left(\sum_\gamma (N_\gamma - 2)\right)\log\det B, \tag{3}$$

where R_n^γ is an instantaneous measure of error of fit:

$$R_n^\gamma = (\mathbf{X}_{n+2}^\gamma - \overline{\mathbf{X}}^\gamma) - A_0(\mathbf{X}_n^\gamma - \overline{\mathbf{X}}^\gamma) - A_1(\mathbf{X}_{n+1}^\gamma - \overline{\mathbf{X}}^\gamma) \tag{4}$$

The nonlinearity introduced by the product terms $A_0\overline{\mathbf{X}}$ and $A_1\overline{\mathbf{X}}$ of R_n^γ can be removed by defining a parameter

$$D^\gamma = (I - A_0 - A_1)\overline{\mathbf{X}}^\gamma \tag{5}$$

so that the instantaneous error becomes $R_n^\gamma = \mathbf{X}_{n+2} - A_0\mathbf{X}_n - A_1\mathbf{X}_{n+1} - D^\gamma$. Object means $\overline{\mathbf{X}}^\gamma$ can always be obtained explicitly if desired, provided $(I - A_0 - A_1)$ is non-singular, by solving for them in equation (5). (Singularity arises in certain special cases, for instance the simple harmonic oscillator, when estimated dynamical parameters are non-unique.) The mean and covariance of the set of object-means $\overline{\mathbf{X}}^\gamma$, $\gamma = 1, \ldots, \Gamma$ are computed using the normal definitions to use as initial mean and covariance for the new state variable $\overline{\mathbf{X}}$ (mean-shape) in the tracking procedure.

Solution

The solution to the maximum likelihood problem is given here but, for space considerations, the proof is omitted. The following set of $\Gamma + 2$ equations is solved simultaneously for the estimated values of parameters A_0, A_1 and D^γ:

$$\begin{aligned} S_2^\gamma - A_0 S_0^\gamma - A_1 S_1^\gamma - D^\gamma(N_\gamma - 2) &= 0, \quad \gamma = 1, \ldots, \Gamma, \\ S_{20} - A_0 S_{00} - A_1 S_{10} &= 0 \\ S_{21} - A_0 S_{01} - A_1 S_{11} &= 0 \end{aligned} \tag{6}$$

where the S_i^γ, S_{ij} are certain moments of the data-set, defined below. This leaves only the random process coupling parameter B to be determined. In fact B cannot be determined uniquely, but any solution of $BB^T = C$, where

$$C = \frac{1}{\sum_\gamma (N_\gamma - 2)}\sum_{\gamma=1}^{\Gamma}\sum_{n=1}^{N_\gamma-2} R_n^\gamma (R_n^\gamma)^T \tag{7}$$

is acceptable. It remains to define the moments of the data-set[3]:

$$S_{ij} = \sum_{\gamma=1}^{\Gamma} \sum_{n=1}^{N_\gamma - 2} \mathbf{X}_{n+i}^{\gamma} (\mathbf{X}_{n+j}^{\gamma})^T - \Gamma \sum_{\gamma=1}^{\Gamma} \frac{S_i^{\gamma} (S_j^{\gamma})^T}{N_\gamma - 2}, \quad i,j = 0,1,2 \tag{8}$$

$$S_i^{\gamma} = \sum_{n=1}^{N_\gamma - 2} \mathbf{X}_{n+i}^{\gamma}, \quad i = 0,1,2. \tag{9}$$

2.3 Results

The extended system with independently modelled class and motion variability has been applied to the agricultural robotics problem described earlier. The outlines of plants are described by B-spline curves parameterised here over a 6-dimensional affine space. The curves are estimated over time in a standard Kalman filter tracker [3]. In figure 4, three different trackers are demonstrated. The first uses a "reasonable" default model, not learnt, but predicting constant velocity and uniform driving noise. The second uses a "joint variability" model in which class and motion variability are lumped together. Because, class variability swamps motion variability for this data (see figure 2), the resulting tracker has unnecessarily weak constraints on temporal shape change. Lastly, we demonstrate a tracker based on the new "independent variability" model which exploits the prior knowledge on shape and motion as thoroughly as possible.

3 Tracking Coupled Objects

Here we explore mechanisms for allowing two coupled objects, one primary and one secondary, to be tracked simultaneously and cooperatively. The aim is to devise an efficient mechanism that applies the coupling approximately but avoids the high computational cost associated with exact coupling.

3.1 Coupled Systems

The configurations of the objects in the coupled system are represented by vectors $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$. General, coupled, linear, stochastic dynamics are represented by:

$$\begin{pmatrix} \mathbf{X}^{(1)} \\ \mathbf{X}^{(2)} \end{pmatrix}_{k+2} = \mathcal{A}_0 \begin{pmatrix} \mathbf{X}^{(1)} \\ \mathbf{X}^{(2)} \end{pmatrix}_{k} + \mathcal{A}_1 \begin{pmatrix} \mathbf{X}^{(1)} \\ \mathbf{X}^{(2)} \end{pmatrix}_{k+1} + B\mathbf{w}_k \tag{10}$$

where $\mathbf{w}$ is a vector of unit normal noise terms coupled into the system via the matrix B and $BB^T = C$, as before. The matrices $\mathcal{A}_0, \mathcal{A}_1$ are damping and elastic coefficients respectively, each composed of submatrices:

$$\mathcal{A}_i = \begin{pmatrix} A_i^{1,1} & A_i^{1,2} \\ A_i^{2,1} & A_i^{2,2} \end{pmatrix}, \; i = 0,1. \tag{11}$$

[3] Moments defined here are rather different from the moments used in [5, 2] despite the similarity of the S_{ij} notation.

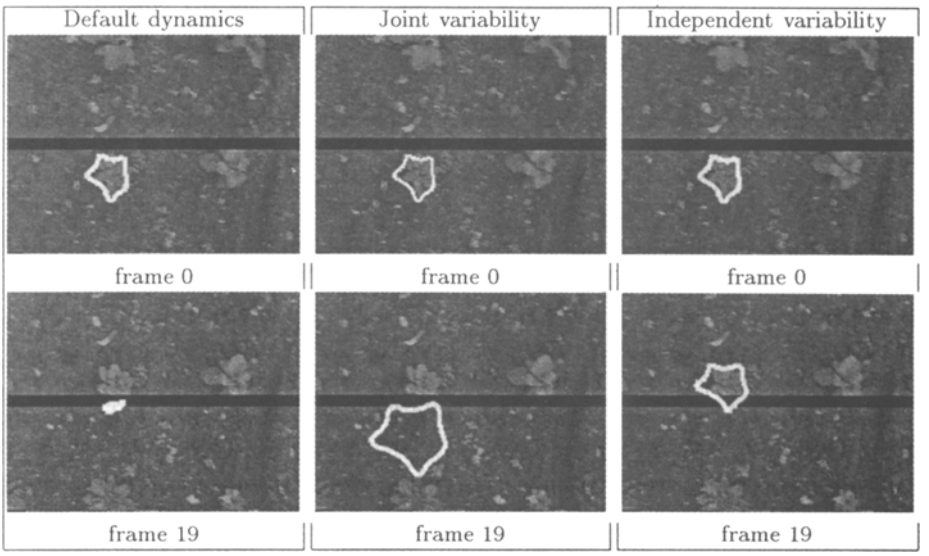

Fig. 4. Joint variability modelling is too weak. *Results are shown here for the three trackers. In this case the tracked plant is rather smaller than average and there is an occluding feature (a simulated shadow). The "joint variability" tracker reverts, after a short time, towards the grand mean of the training set and hence loses lock on the plant. The "independent variability" tracker however adjusts rapidly, via its additional mean-configuration variable* $\overline{\mathrm{X}}$*, to the small size of this plant and continues to track successfully.*

Each submatrix contains coefficients either for one of the objects or for the cross-couplings between objects.

Frequently, as with head and lip motion, one object $\mathbf{X}^{(1)}$ is primary and drives a secondary object $\mathbf{X}^{(2)}$. The motion of the primary object is assumed independent of the secondary, and is modelled stand-alone so that

$$A_i^{1,2} = 0, \; i = 0, 1. \tag{12}$$

However, the motion of the secondary system $\mathbf{X}^{(2)}$ has two components, one caused by the primary system, $\mathbf{X}^{(1)}$, and one caused by its own independent motion, $\mathbf{Y}$, with dynamics given by

$$\mathbf{Y}_{k+2} = A_0^{2,2}\mathbf{Y}_k + A_1^{2,2}\mathbf{Y}_{k+1} + B^{(2)}\mathbf{w}_k^{(2)} \tag{13}$$

and a direct coupling is added to represent the influence of the primary process:

$$\mathbf{X}^{(2)} = \mathbf{Y} + \mu\mathbf{X}^{(1)}, \tag{14}$$

where μ is a constant matrix of the appropriate dimension. Combining (13) and (14) and comparing with (10) leads to constraints on the dynamical parameters:

$$A_i^{2,1} = \mu(A_i^{1,1} - A_i^{2,2}) \; i = 0, 1. \tag{15}$$

3.2 Learning Coupled Dynamics

With the constraints (12) and (15), the maximum likelihood learning algorithm turns out to be nonlinear, with a consequent need for iteration and the possibility of multiple solutions. To avoid this we propose that (15) be relaxed, retaining only (12). (This can be interpreted physically as allowing a velocity coupling in addition to the positional coupling (14).) The resulting learning algorithm is now linear.

Given its isolation, coefficients for the motion of the primary object are learned exactly as for a single object, as earlier and in [5]. Dynamics of the secondary object and its coupling to the primary are obtained (proof omitted here) by solving the following simultaneous equations:

$$\begin{aligned}
S_{20}^{(2,2)} - A_0^{2,2} S_{00}^{(2,2)} - A_1^{2,2} S_{10}^{(2,2)} - A_0^{2,1} S_{00}^{(2,1)} - A_1^{2,1} S_{01}^{(2,1)} &= 0 \\
S_{21}^{(2,2)} - A_0^{2,2} S_{01}^{(2,2)} - A_1^{2,2} S_{11}^{(2,2)} - A_0^{2,1} S_{10}^{(2,1)} - A_1^{2,1} S_{11}^{(2,1)} &= 0 \\
S_{20}^{(2,1)} - A_0^{2,2} S_{00}^{(2,1)} - A_1^{2,2} S_{10}^{(2,1)} - A_0^{2,1} S_{00}^{(2,2)} - A_1^{2,1} S_{01}^{(2,2)} &= 0.
\end{aligned}$$

where moments are defined:

$$S_{ij}^{(p,q)} = \sum_{n=1}^{m-2} \mathbf{X}_{n+i}^{(p)} \mathbf{X}_{n+j}^{(q)}, \quad i,j = 0,1,2, \quad p,q = 1,2.$$

(The algorithm given here is for the case of known mean configuration $\overline{\mathbf{X}}^{(1)} = 0, \overline{\mathbf{X}}^{(2)} = 0$.) The covariance C is calculated from instantaneous error R_n as earlier and [5].

3.3 Computational Complexity for Filtering

In the visual tracking application, the above dynamical system is used as a predictor in a Kalman filter [7, 1] for curve tracking — a dynamic contour tracker [15, 3]. In the filter, the prediction step covariance $\mathcal{P}$ evolves, in the conventional manner, as:

$$\mathcal{P}_{k+1|k} = \begin{pmatrix} \mathcal{A}_{1,1} & 0 \\ \mathcal{A}_{2,1} & \mathcal{A}_{2,2} \end{pmatrix} \begin{pmatrix} P_{1,1} & P_{1,2} \\ P_{1,2}^T & P_{2,2} \end{pmatrix}_{k|k} \begin{pmatrix} \mathcal{A}_{1,1}^T & \mathcal{A}_{2,1}^T \\ 0 & \mathcal{A}_{2,2}^T \end{pmatrix} + C,$$

a critical step from the point of view of computational load. Even though $\mathcal{A}_{1,2} = 0$ and $\mathcal{A}_{2,1}$ is sparse, the covariance $\mathcal{P}_{k+1|k}$ remains dense, leaving unabated the full computational load of $6\alpha n^3$ operations, where α is a constant and n is the filter's dimension. This compares with αn^3 operations to track an isolated object, or $2\alpha n^3$ for two uncoupled objects. So the price of representing the coupling faithfully is a 3-fold increase in computation.

The measurement process suffers from similar costs and as before, the amount of computation to run coupled systems is $6\gamma n^3$ where γn^3 is the amount of computation required for an isolated object. Again, coupling incurs a 3-fold cost increase.

3.4 Approximating the Coupling

We refer to the full solution of the Kalman filtering problem as "strongly coupled". A "weakly coupled" approximation to the full algorithm is constructed by treating the state of the primary system as if it were known exactly, for the purpose of determining its effect on the secondary system. In other words, equation (14) is replaced by

$$\mathbf{X}^{(2)} = \mathbf{Y} + \mu\hat{\mathbf{X}}^{(1)}$$

where $\hat{\mathbf{X}}^{(1)}$ is the *estimate* of the primary system's state. The effect of the primary system on the secondary becomes entirely deterministic. Covariance terms $P_{1,1}$ and $P_{2,2}$ become mutually independent and, moreover, $P_{1,2} = P_{2,1}^T = 0$. Covariances must be computed continuously to maintain the Kalman gain matrices, so this is where the weak coupling allows substantial savings in computation. Of course, the restricted P matrices computed in this way cannot be claimed to approximate the true covariances, and since they are used to define Kalman gains, those gains are substantially altered by the approximation. However, these are open loop gains and it is well known large changes in open loop gain often cause only small perturbations in closed loop response. This explains somewhat the accuracy of results with the weak coupling reported below.

Weak coupling can be generalised to larger numbers of objects tracking simultaneously, with or without object hierarchy. In general, if a single filter has a computational burden of Γ, then, in the strongly coupled case, with m objects, a burden of $\Gamma(m^3 + m^2)/2$ is incurred. In the weakly coupled case it is reduced to Γm.

3.5 Head and Lip Tracking

Previously Moses *et al* [12] designed a contour tracker to follow an intensity valley between the lips. The valley proved to be a robust feature under a wide range of lighting conditions but because of the extreme aspect ratio of the mouth, the tracker tended to be unstable to horizontal translation (figure 3). The instability problem is tackled here by a weak coupling between the lip as secondary object to the head. A five dimensional space was used to model the head outline, comprising translation in the x and y direction, uniform scaling, and rotation about the vertical axis (using two degrees of freedom not one, to allow for silhouette effects).

Figure 5 shows the effect of tracking when coupling is enabled. In this case the strongly coupled algorithm was run. The coupling is such that, appropriately enough, the horizontal motion of the head and its rotation affect the horizontal location of the lips. The size of the head also affects the size of the lips, allowing for zoom-coupling. The figure clearly demonstrates that coupling is effective. Figure 6 shows that weak coupling produces results similar to strong coupling, and that without coupling, the tracker fails at an early stage. Figure 7 displays lip deformations as a function of time for strong and weak coupling. Two affine components of motion are plotted: horizontal and vertical scaling. The most obvious feature is that, as before, the uncoupled tracker loses lock after about

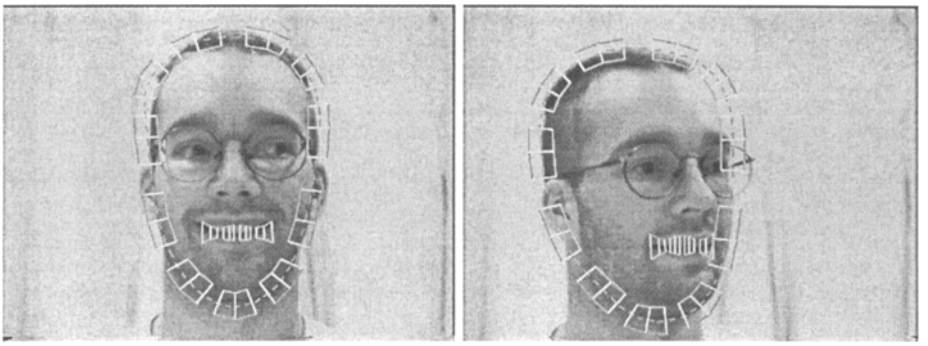

Fig. 5. Strongly-coupled algorithm is successful. *This set of images shows that the poor tracking performance obtained from an isolated mouth tracker (figure 3) is substantially redeemed by appropriate coupling to head motion.*

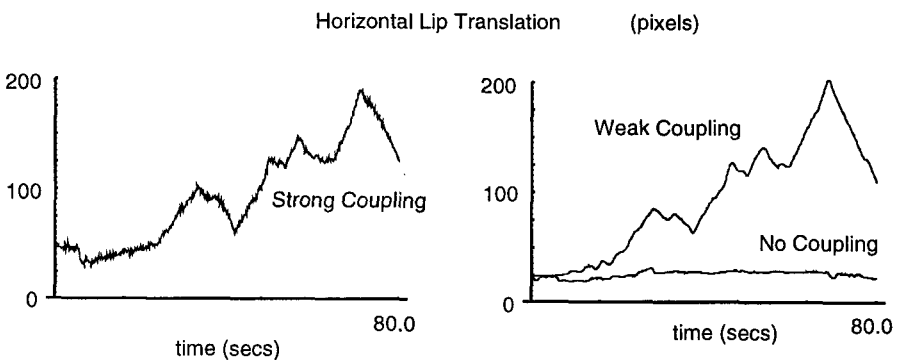

Fig. 6. Lip tracking is successful only when coupled to the head tracker. *The first graph displays lip translation as a function of time, in the case of strong coupling. An independent experiment has confirmed that this result is broadly accurate. The second graph shows that without coupling the tracker fails at an early stage but that with weak coupling accurate tracking is obtained.*

20 seconds. It is also clear that weak coupling produces smoother estimates of the lip shape than strong coupling.

4 Conclusions

Work presented here reinforces the growing acceptance that careful modelling of object shape, motion and environment is crucial to effective performance of visual processes in general and visual tracking in particular. Stochastic differential equations are particularly useful as models for this purpose because they are the basis of prediction in the Kalman filter. Moreover, procedures exist for learning such models from examples.

More specifically, stochastic modelling for tracking has been advanced in two particular regards in this paper. The first is in clearly separating object vari-

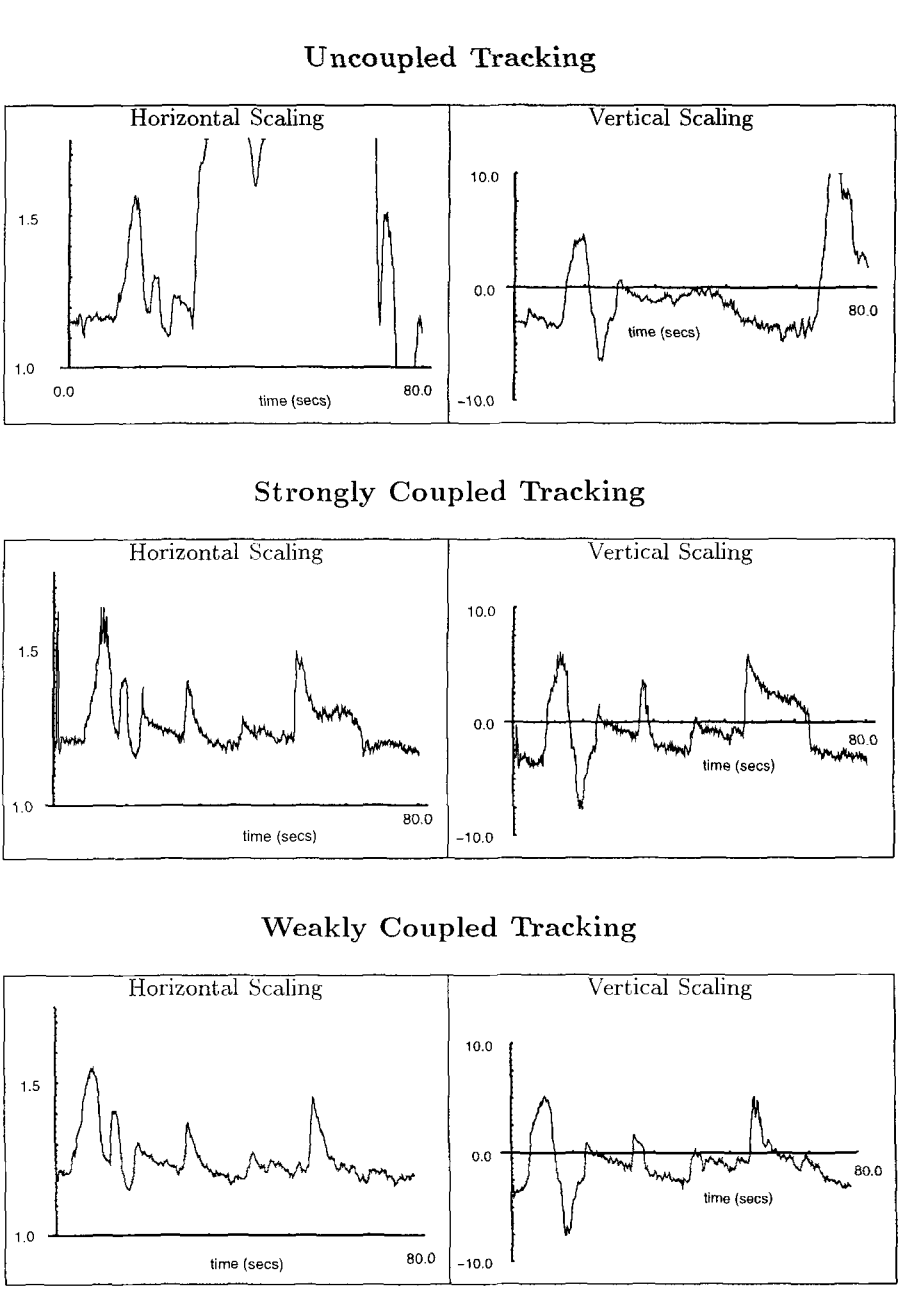

Fig. 7. Comparative performance of tracking algorithms. *This figure shows two affine components of the lip motion as a function of time. The top pair of graphs show how lock is lost very early in the sequence (after about 20.0 sec) in the absence of any coupling. The final pair shows that when the coupling is weak, performance is close to that in the strongly coupled case.*

ations arising within a class of objects from those that occur temporally, during motion. The second is in building coupled motion models to track objects whose motions are not independent, for instance head and lips. In both case, appropriate modelling leads to substantial enhancements of tracking performance.

References

1. Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association.* Academic Press, 1988.
2. A. Baumberg and D. Hogg. Generating spatiotemporal models from examples. In *Proc. BMVC*, 413–422, 1995.
3. A. Blake, R. Curwen, and A. Zisserman. A framework for spatio-temporal control in the tracking of visual contours. *Int. Journal of Computer Vision*, 11(2):127–145, 1993.
4. A. Blake, M. Isard, and D. Reynard. Learning to track the visual motion of contours. *Artificial Intelligence*, 78:101–133, 1995.
5. A. Blake and M.A. Isard. 3D position, attitude and shape input using video tracking of hands and lips. In *Proc. Siggraph*, 185–192. ACM, 1994.
6. T.F. Cootes, C.J. Taylor, A. Lanitis, D.H. Cooper, and J. Graham. Buiding and using flexible models incorporating grey-level information. In *Proc. 4th Int. Conf. on Computer Vision*, 242–246, 1993.
7. A. Gelb, editor. *Applied Optimal Estimation.* MIT Press, Cambridge, MA, 1974.
8. D.B. Gennery. Visual tracking of known three-dimensional objects. *Int. Journal of Computer Vision*, 7:3:243–270, 1992.
9. C.G. Goodwin and K.S. Sin. *Adaptive filtering prediction and control.* Prentice-Hall, 1984.
10. C. Harris. Tracking with rigid models. In A. Blake and A. Yuille, editors, *Active Vision*, 59–74. MIT, 1992.
11. D.G. Lowe. Robust model-based motion tracking through the integration of search and estimation. *Int. Journal of Computer Vision*, 8(2):113–122, 1992.
12. Y. Moses, D. Reynard, and A. Blake. Determining facial expressions in real time. In *Fifth International Conference on Computer Vision*, 296–301, Massachusetts, USA, 1995.
13. A. Pentland and B. Horowitz. Recovery of nonrigid motion and structure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13:730–742, 1991.
14. R.S. Stephens. Real-time 3D object tracking. *Image and Vision Computing*, 8(1):91–96, 1990.
15. R. Szeliski and D. Terzopoulos. Physically-based and probabilistic modeling for computer vision. In B. C. Vemuri, editor, *Proc. SPIE 1570, Geometric Methods in Computer Vision*, 140–152, San Diego, CA, July 1991. Society of Photo-Optical Instrumentation Engineers.
16. D. Terzopoulos and R. Szeliski. Tracking with Kalman snakes. In A. Blake and A. Yuille, editors, *Active Vision*, 3–20. MIT, 1992.

Acknowledgement: We are grateful for the support of the BBSRC and the Rhodes Trust.