TRACKING OF MOVING OBJECTS WITH MULTIPLE MODELS USING GAUSSIAN MIXTURES

Jorge S. Marques

IST / ISR
Torre Norte, Av. Rovisco Pais,
1096 Lisbon, Portugal

João M. Lemos

IST / INESC
R. Alves Redol 9
1000 Lisboa, Portugal

ABSTRACT

This paper addresses the problem of tracking of objects with complex shape and motion dynamics. The approach followed relies on multiple models based on Gaussian mixtures and hidden Markov models. A tracking algorithm derived from Nonlinear Filtering is presented and illustrated in two situations. In the first, two points moving independently along a line are tracked, only one being observed at each time. In the second, two dimensional objects are tracked, under severe shape deformations. Unlike other multi-model approaches, the proposed method relies on parametric techniques providing an efficient tool to update shape and motion estimates.

1. INTRODUCTION

Algorithms for solving engineering problems relying on multiple models are increasingly finding applications in Signal Processing and Control. This stems from the difficulties inherent to a reality (be it a process plant, a speech signal or the image of a changing object) which is too difficult to be modeled by a single model, simple enough to be managed. Further, if the reality to be modeled undergoes transformations as time passes, hidden Markov models (HMM) [8] are a possible way of taking this into account. Despite HMM's present limitations (e. g. they poorly model the influence of the remote past), they have the merit of leading to algorithms with tractable complexity.

In Control, one can think of the problem of a process working under a slowly varying operation condition. See [7] for a recent reference among several others possible. In Image Processing, image sequences of moving objects with time-varying shape provide another class of examples. For example, a car passing in a highway in front of a fixed camera yields different shapes according to, say, the car is far away and the front is mostly seen, or it is close and significant parts of the side appear. A moving hand making signs according to its shape provides another example. In both cases, the images captured have a time varying structure, each of the possibilities corresponding to a different model in an alphabet.

This paper addresses the problem of tracking moving objects with abrupt changes of shape or motion. The approach followed relies on multiple models based on Gaussian mixtures and hidden Markov models. A tracking algorithm derived from Nonlinear Filtering is presented. This algorithm is inspired in the work of Tugnait on the detection and estimation of abruptly changing systems [11]. Unlike other approaches (e.g. the Condensation method recently described in [2]) the proposed algorithm relies on a parametric model and provides an efficient tool to update the a posteriori density even if the number of parameters is high. In order to illustrate this algorithm, two situations are considered. The first situation is a synthetic example in order to highlight the main features of the approach. It consists of the tracking of two points independently moving along a line, only one being observed at each time. In the second, a two dimensional object with a changing shape is tracked in a sequence of images.

2. PROBLEM STATEMENT

A simple model will be adopted to describe the shape of a moving object with abrupt motion or shape changes. Let $M_1, \ldots, M_M$ be M shape matrices associated to M views of a deformable object. Each view is defined as a sequence of boundary points or as a spline curve with known control points. Let $o = (x_1, \ldots, x_N, y_1, \ldots, y_N)$ be the coordinates of N features detected in the image (e.g., edge points) associated with N model points. It will be assumed in the sequel that o is an affine transform of one of the shape matrices, corrupted by white Gaussian noise, i.e., (see the details in [6])

$$o_t = C_{q_t} a_t + v_t \qquad (1)$$

where a_t is a vector of affine parameters, and its first derivatives, $q_t \in \{1, \ldots, M\}$ is a model label, v_t is a noise vector with multivariate normal distribution, $N(0, R)$ and

$$C_{q_t} = [BM_{q_t} \;\; 0] \qquad (2)$$

is the observation matrix, B being an interpolation matrix in the case of B-spline models, and the identity matrix in the case of point models. It is assumed that a_t is a random process described by a stochastic difference equation and the label sequence q_t is a Markov chain.

Problem

Given a set of observations $O^t = (o_1, \ldots, o_t)$, detected in a sequence of t images, we wish to estimate the motion (affine) parameters a_t and the model label q_t.

This is a nonlinear filtering problem [3]. If the joint probability density function conditioned on the observations $p(a_t, q_t / O^t)$ is evaluated, the unknown parameters can be estimated in a number of ways, e.g., using the MAP method

$$(\hat{a}_t, \hat{q}_t) = \arg\max_{a_t, q_t} p(a_t, q_t / O^t) \quad (3)$$

It will be assumed that a_t, q_t contain all the information about the past needed to generate future observations. Since a_t is a continuous random variable and q_t is discrete, (a_t, q_t) is denoted as a mixed-state.

If the model estimate, $\hat{q}_t$, is not needed, the pose parameters can be obtained by the maximization of the a posteriori density conditioned on the observations, $p(a_t/O^t)$. This function is closely related to the joint density of the mixed-state variables

$$p(a_t/O^t) = \sum_{q_t} p(a_t, q_t/O^t) \quad (4)$$

The computation of $p(a_t/O^t)$ is addressed in the next section assuming that a_t is the output of a switched bank of filters. This allows to express the optimal state distribution as a Gaussian mixture with an (exponentially) increasing number of components and provides a tree structure to compute the unknown parameters. A recursive method is described to provide practical algorithms to perform these computations. The number of mixture components increases with t. To overcome this difficulty, the optimal distribution is approximated by a Gaussian mixture with a smaller number of modes. Strategies for complexity reduction are addressed in section 4.

3. PROPAGATION OF CONDITIONAL DENSITY

Let us first address the estimation of $\hat{a}_t$, assuming that the model sequence Q^t is known. In the sequel we shall assume that the state vector a_t is the output of a first order random equation

$$a_t = A_{q_t} \; a_{t-1} + w_t \quad (5)$$

where the characteristic polynomial of matrix A is strictly Hurwitz and w_t is a white vector with normal distribution $N(0, Q)$, uncorrelated with the observation noise v_t. The initial condition is also a random variable with normal distribution $N(a_0, P_o)$. Equations (5,1) define a stochastic linear system. The estimation of the affine parameters, a_t, from the present and past observations, O^t, is a well known state estimation problem: the distribution of a_t given O^t is $N(\hat{a}_t, P_t)$, with $\hat{a}_t, P_t$ updated by the Kalman filter equations. The MAP state estimate is $\hat{a}_t$.

When the model sequence is unknown, we have to compute $p(a_t, Q^t/O^t)$ for all admissible sequences and obtain $P(a_t/O^t)$ as the marginal density of this joint probability distribution:

$$p(a_t/O^t) = \sum_{Q^t} p(a_t, Q^t/O^t) = \sum_{Q^t} c_{Q^t} p(a_t/Q^t, O^t) \quad (6)$$

where $Q^t = (q_1, \ldots, q_t)$ is a model sequence and $c_{Q^t} = P(Q^t/O^t)$. The density $p(a_t/Q, O^t)$ is a normal density function with mean and covariance updated by Kalman filtering as discussed before. Therefore $p(a_t/O^t)$ is a mixture of Gaussians, each of them being associated to a specific model sequence Q^t. Since all model sequences $Q^t \in$

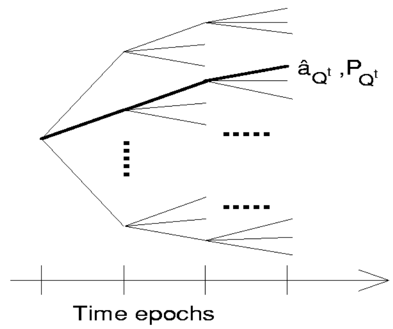

Figure 1: Computation of the mixture components

$\{1, \ldots, M\}^t$ are allowed, the mixture will have M^t components. For example, if the observations are generated by two models, the mixture will have 8 components at t=3.

The computation of the mean and covariance of each component is organized in a tree structure where each branch corresponds to an iteration of a Kalman filter (see fig. 1). This optimal structure cannot be directly implemented and some complexity reduction schemes have to be devised to avoid the combinatorial explosion. This problem is addressed in section 4.

Sometimes, it is enough to assume that a single transition is allowed during a given time interval. In this case, most of the tree paths are discarded and the number of nodes increases linearly with the length of the interval. A more drastic attitude is even adopted in control, see e. g. [7], by assuming that the data is generated by the same model in a sliding time interval. These approaches will not be pursued here: Multi-model tracking will be addressed with full generality allowing all possible transitions.

Consider now the computation of the mixing parameters $c_{Q^t} = P(Q^t/O^t)$. If the model sequence is a Markov chain with known transition probabilities, T_{ij}, the mixing parameters can be recursively updated by [6]

$$c_{Q^t} = kT_{q_{t-1}q_t} G(o_t) c_{Q^{t-1}} \quad (7)$$

where G is a Gaussian density function and k is a constant, obtained from the normalization condition $\sum c_{Q^t} = 1$. The algorithm used to compute each tree node is summarized in table 1. The following example illustrates the use of switched models and the performance of the proposed algorithm with synthetic data.

Example

Consider two bees flying on a line with independent random motions (Fig. 2a). Suppose there is a sensor which provides the coordinate of one of the bees (we do not know which) at each instant of time and the measurement is corrupted by noise. We wish to estimate the location of both insects at every instant of time from these noisy observations.

This is a state estimation problem with multiple models. The state vector $a = (x_1, x_2)$ contains the bees coordinates. It will be assumed that the bee motion is the output of a

For each node created at instant t:

i) update mean and covariance

Prediction	*Filtering*
$\hat{a}^- = A_{q_t}\hat{a}'$	$\hat{a} = \hat{a}^- + K(o_t - C_{q_t}\hat{a}^-)$
$P^- = A_{q_t}P'A_{q_t}^T + Q$	$K = P^- C_{q_t}^T (C_{q_t} P^- C_{q_t}^T + R)^{-1}$
	$P = (I - KC_{q_t})P^-$

ii)update mixing parameters

$$c_{Q^t} = kT_{q_{t-1}q_t} G(o_t) c_{Q^{t-1}}$$

where

$$(\hat{a}, P) \triangleq (\hat{a}_{Q^t}, P_{Q^t}), \quad (\hat{a}', P') \triangleq (\hat{a}_{Q^{t-1}}, P_{Q^{t-1}})$$

Table 1: Tracking algorithm

stochastic equation

$$a_t = \begin{bmatrix} .98 & 0 \\ 0 & .5 \end{bmatrix} a_{t-1} + w_t \tag{8}$$

with $w_t \sim N(0, Q)$, $Q = diag(1, 9)$ and the sensor equations are

$$\begin{array}{ll} model1 & o_t = [1 \;\; 0] a_t + v_t \\ model2 & o_t = [0 \;\; 1] a_t + v_t \end{array} \tag{9}$$

with $v_t \sim N(0, .1)$. The model transitions are described by a Markov chain with transition matrix

$$T = \begin{bmatrix} .9 & .1 \\ .1 & .9 \end{bmatrix} \tag{10}$$

Figure 2 shows the output of the tracking algorithm described in table I for a single experiment. Figure 2b shows the observations available to locate both bees (try to guess their motion from this information). Fig. 2e shows which bee is measured at each instant of time (this information is displayed to give insight on the problem but it is not used to estimate the bee trajectories). The true bee trajectories and the MAP estimates are displayed in Fig. 2c,d. This experiment shows that it is possible to locate both insects most of the time, i.e., the algorithm manages to guess which insect is being sensed. Finally, Figs. 2f-h show the density function $p(a_t/O^t)$ (Gaussian mixture) at specific time instants, identified by dots in Fig. 2e.

If one of the variables is observed for a long period, its estimate has a small variance while the variance of the other variable increases. This effect is clearly noticed in Fig. 2f-h. Compare, for instance, figs 2g and 2h. In fig. 2g the observations model has just moved from a situation in which bee 1 is being observed, to a situation in which bee 2 is observed. Thus, the spread of the density along axis 1 is much smaller than along axis 2. After some time passes (fig. 2h) the algorithm "recognizes" that bee 2 is being observed and narrows its uncertainty about it. Conversely, the density along the bee 1 axis will spread.

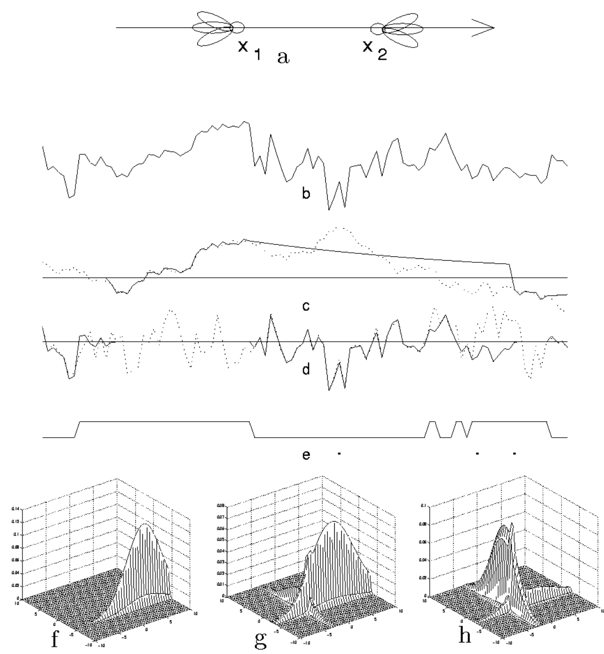

Figure 2: Finding the bees: a) what we want to know; b) observations; c) bee trajectories (dashed line) and the MAP estimates (solid line); and d) model sequence used to generate the observations (not used for estimation); f-h) state densities at specific time instants (see marks in c)

4. APPLICATION TO TRACKING

Two difficulties have to be tackled for applying the previous algorithm to tracking: the combinatorial explosion of the number of components and the detection of the visual features (observations). These aspects will be briefly addressed in the sequel.

4.1. Complexity Reduction

In practice, the number of components of the Gaussian mixture cannot grow to infinity and must be limited. Several strategies have been proposed [11]. In this paper, two methods are used to achieve this goal: component elimination and merging. The first method discards components with mixing parameters smaller than a given threshold (e.g., 10^{-3}). These components produce a negligeable contribution to the mixture density. The second method tries to avoid multiple components with identical densities by merging then into a single component. The Kullback divergence is used to decide if two components are similar (the divergence is computed for all pairs of components; the pair with smallest divergence is merged if the divergence is bellow a given threshold; the process continues until there is no pair meeting the merging conditions). The divergence between two normal distributions, $N(\mu, P), N(\mu', P')$ is given by [9]

$$D = \tfrac{1}{2}(\mu - \mu')^T (P^{-1} + P'^{-1})(\mu - \mu') + \tfrac{1}{2} tr\{P^{-1}P' + P'^{-1}P - 2I\} \tag{11}$$

A similar criterion is used in [5] to approximate a periodic function by a Gaussian mixture in the context of nonlinear phase estimation.

4.2. Feature Detection

In this paper it is assumed that the shape model is attracted by feature points detected in the image as in active contours [4]. Several methods are available for detecting image features, e.g., by using line searching along the normal directions at specific contour points [10] or by computing the data centroids using competitive learning methods [1]. In all of these, feature detection requires a shape estimate. Therefore, the tracking algorithm has a feedback loop: the shape model tries to track the image features in a video sequence but the detected features depend on the predicted shape. This is an old problem. What is new is the use of multiple models which brings an additional difficulty: the handling of multiple predictions $\hat{o}^- = C_i\hat{a}^-$ based on different shape models. In this paper, only the most probable model is used in feature extraction.

5. RESULTS

The proposed algorithm was used to track moving objects with significant shape changes in video sequences. A simple dynamic model was adopted to describe the evolution of the motion parameters: the derivative of each parameter is modeled by a Wiener process, leading to a dynamic equation

$$a_t = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix} a_{t-1} + w_t \qquad (12)$$

where $a_t \in R^{12}$ is a state vector containing the motion parameters and their derivatives, I is the identity matrix, and 0 the null matrix. It was assumed that the object shape in the first image is known. In each new image the following operations are performed: i) shape prediction according to $\hat{o}^- = C_i\hat{a}^-$ (i being the most probable model); ii) feature detection by line search along the normal directions to the contour; iii) update of mixture components according to table I; iv) component reduction as described in section 4 and v) parameter estimation using the MAP method.

Figure 3 shows tracking results obtained with a sequence of a moving hand. Three shape models were used as shown in fig. 3a. Fig. 3b-c displays the selected shape model transformed using the estimates of the affine parameters. This example illustrates the ability of the proposed algorithm to cope with significant shape deformation, keeping good tracking capability. Only parametric models as the one described in this paper allow to efficiently propagate density functions in a 12D space.

6. CONCLUSIONS

This paper addresses the problem of tracking of moving objects with significant changes of shape or motion dynamics. The approach followed relies on multiple models based on Gaussian mixtures and hidden Markov models. A tracking algorithm derived from Nonlinear Filtering is presented and illustrated in two situations. In the first, two points moving independently along a line are tracked, only one being observed at each time. In the second, a two dimensional object with a changing shape is tracked. Unlike other approaches the proposed method relies on a parametric model of the a posteriori distribution providing efficient tools to update the parameter estimates.

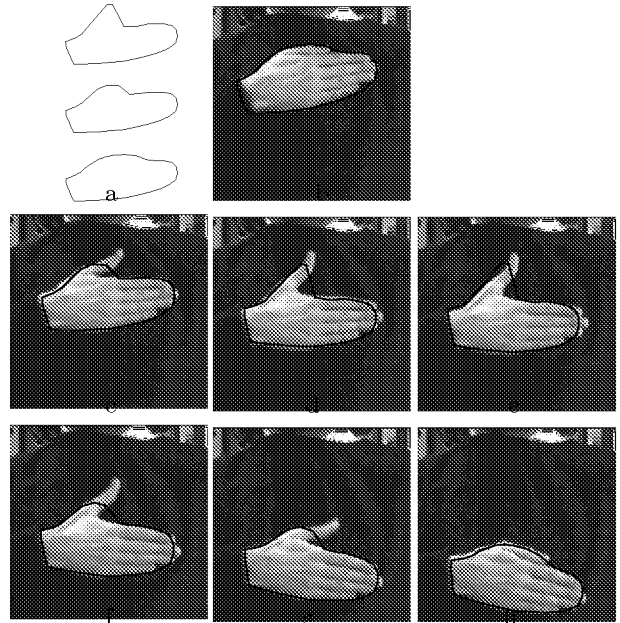

Figure 3: Tracking results with real data. a) shape models b-h) shape estimates

7. REFERENCES

[1] A. Abrantes, J. Marques, A Class of Constrained Clustering Algorithms for Object Boundary Detection, IEEE Trans. on Image Processing, 5, 1507-1521, 1996.

[2] M. Isard, A. Blake, A Mixed-State Condensation Tracker with Automatic Model-Switching, Proc. Int. Conf. Computer Vision, 1998.

[3] Jazwinski, Stochastic Processes anf Filtering Theory, Academic Press, 1970.

[4] M. Kass, A. Witkin, D. Terzopoulos, Snakes: Active Contour models, International Journal of Computer Vision,1, 321-331, 1987.

[5] J. Leitão, J. Moura, Nonlinear Phase Estimators-Based on the Kullback Distance, Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, pp. 521-524, Adelaide, 1994.

[6] J. Marques, J. Lemos, Tracking of Moving Objects Based on a Mixture of Gaussians,int. report, IST, 1998.

[7] A. S. Morse, Supervisory control of families of linear set-point controllers – Part 1: Exact matching, IEEE Trans. Automat. Contr., 41, 1413-1431, 1996.

[8] L. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proc. IEEE, 77, 2, 257-286.

[9] B. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, 1996.

[10] D. Terzopoulos and Szeliski, Tracking with Kalman Snakes, in Active Vision, Blake and Yuille, eds., MIT Press, 1992

[11] J. Tugnait, Detection and Estimation for Abruptly Changing Systems, Automatica, 606-615, 1982.