

A Maximum A Posteriori Probability Viterbi Data Association Algorithm for Ball Tracking in Sports Video

Fei Yan William Christmas Josef Kittler
CVSSP, University of Surrey, Guildford, Surrey, GU2 7XH, UK
{f.yan, w.christmas, j.kittler}@surrey.ac.uk

Abstract

In this paper, we derive a data association algorithm for object tracking in a maximum a posteriori framework: the output of the algorithm is the sequence of measurement-to-target associations with maximum a posteriori probability. We model the object motion as a Markov process, and solve this otherwise combinatorially complex problem efficiently by applying the Viterbi algorithm. A method for combining forward and backward tracking results is also developed, to recover from tracking errors caused by abrupt motion changes of the object. The proposed algorithm is applied to broadcast tennis video to track a tennis ball. Experiments show that its performance is comparable to that of a computationally more expensive particle-filter-based algorithm.

1. Introduction

In automatic annotation of sports video, higher-level descriptions generally rely on low-level features. In the context of ball game, tracking of the ball is essential for the annotation. Several ball tracking algorithms have been reported in the literature. The object being tracked includes base ball [7], tennis ball [3, 4, 5], soccer ball [9], *etc.* However, most of these algorithms are nearest neighbour type of simple tracker. By making single hypothesis, the more difficult data association problem is avoided, but at a price of loss of robustness. On the other hand, state-oriented tracking methods that try to solve the data association problem implicitly, *e.g.*, Probabilistic Data Association Filter (PDAF), is not appropriate for ball tracking, because the estimated state is “contaminated” by clutter-originated measurements, and thus not accurate enough for annotation.

Unlike the techniques mentioned above, we explicitly address the data association problem encountered in the tracking of the ball in broadcast sports video. This problem is challenging due to severe occlusion, heavy background clutter and abrupt motion changes of the ball. In

this paper, we propose a Viterbi-based data association algorithm to tackle these difficulties. The proposed algorithm seeks the sequence of measurement-to-target associations with maximum *a posteriori* probability, among all possible sequences of associations. This combinatorially complex problem is solved efficiently by applying the Viterbi algorithm. A method for combining forward and backward tracking results is also introduced, which is used to recover from tracking errors caused by abrupt motion changes of the ball. The proposed algorithm is applied to broadcast tennis video to track a tennis ball. Experiments show that its performance is comparable to that of a computationally more expensive particle-filter-based algorithm.

The rest of this paper is organised as follows: the proposed algorithm is described in Section 2. Its performance is presented in Section 3. Conclusions are given in Section 4.

2. The Algorithm

Assume we have a ball candidate generation module, and several candidates are produced by this module in each field. Note that these candidates may contain false positives that have originated from clutter, and may not contain the true object, due to occlusion, misclassification, *etc.* Now the tracking is essentially a data association problem: determine which, if any, of these candidates is the object. In [6], the object motion is assumed to be a Markov process, and the Viterbi algorithm is adopted to solve the data association problem in a maximum likelihood framework. We refer to the algorithm in [6] as maximum likelihood Viterbi data association (ML-VDA). In this paper, the same Markov assumption is made, and a maximum *a posteriori* probability Viterbi data association algorithm (MAP-VDA) is derived.

2.1 MAP-VDA

We model the motion of the ball by a dynamic system with a system equation $\mathbf{x}_k = A\mathbf{x}_{k-1} + \mathbf{v}$ and an observation equation $\mathbf{z}_k = H\mathbf{x}_k + \mathbf{w}$, where $\mathbf{x}_k \in \mathbb{R}^4$ is the

true state (position and velocity of the ball), $\mathbf{z}_k \in \mathbb{R}^2$ is the measurement (position of a candidate), \mathbf{v} , \mathbf{w} are zero-mean white Gaussian noise, and A , H are the state transition matrix and observation matrix, respectively. In the absence of measurement origin uncertainty, a least square estimate of the state is provided by the classical Kalman filter (KF). When measurement origin uncertainty is present, however, the data association problem has to be dealt with before the estimation problem.

Assume in the k^{th} field of the sequence, m'_k candidates are generated, and are denoted by $C(k) = \{c_i(k)\}_{i=1}^{m'_k}$. The total number of sequences of measurements at time k is then $L_k = \prod_{j=1}^k (m'_j + 1)$, where the extra 1 corresponds to the case that the ball is not detected in field j . Let us denote the l^{th} sequence of measurements at time k by $Z^{k,l} \triangleq \{Z^{k-1,s}, \mathbf{z}_{i_{k,l}}(k)\}$, where $Z^{k-1,s}$ is the s^{th} sequence of measurements at time $k-1$, $\mathbf{z}_{i_{k,l}}(k)$ is the measurement at time k that is in the l^{th} sequence of measurements. We define $\theta^{k,l}$ to be the event that $Z^{k,l}$ is the correct sequence of measurements up to time k . Clearly, $\theta^{k,l}$ can be decomposed as $\theta^{k,l} \triangleq \{\theta^{k-1,s}, \theta_{i_{k,l}}(k)\}$, where $\theta^{k-1,s}$ is the event that $Z^{k-1,s}$ is the correct sequence of measurements up to time $k-1$, and $\theta_{i_{k,l}}(k)$ is the event that $\mathbf{z}_{i_{k,l}}(k)$ is the correct measurement at time k , *i.e.*, the event that candidate $c_{i_{k,l}}(k)$ is the object. We also define Z^k to be the set of sets of measurements up to time k , where $Z^k \triangleq \{Z(k), Z^{k-1}\}$, Z^{k-1} is the set of sets of measurements up to time $k-1$, and $Z(k)$ is the set of measurements at time k . According to Bayes' theorem,

$$\begin{aligned} \beta^{k,l} &\triangleq P\{\theta^{k,l}|Z^k\} = P\{\theta_{i_{k,l}}(k), \theta^{k-1,s}|Z(k), Z^{k-1}\} \\ &\propto P\{Z(k)|\theta_{i_{k,l}}(k), \theta^{k-1,s}, Z^{k-1}\} \\ &\quad \times P\{\theta_{i_{k,l}}(k)|\theta^{k-1,s}, Z^{k-1}\} \times \beta^{k-1,s} \end{aligned} \quad (1)$$

Eq. 1 shows that $\beta^{k,l}$ can be computed recursively. Once $\beta^{k,l}$ is computed, \mathbf{x}_k can be estimated as $\hat{\mathbf{x}}(k|k) = \sum_{l=1}^{L_k} \hat{\mathbf{x}}^l(k|k)\beta^{k,l}$, where $\hat{\mathbf{x}}^l(k|k)$ is the estimate given by the Kalman filter that is associated with the l^{th} sequence of measurements. This algorithm is known as the optimal Bayesian filter (OBF): the optimality is achieved at a price of exponentially increasing complexity. In practice, the probabilistic data association filter (PDAF) has been widely adopted as a suboptimal Bayesian solution [1]. Unlike the OBF and the PDAF, which are state-oriented, we propose an association-oriented algorithm. Our goal is to find the sequence of measurements with maximum *a posteriori* probability, *i.e.*, we seek the event

$$\theta^{k,l^*} \triangleq \arg \max_{\theta^{k,l}} \beta^{k,l} = \arg \max_{\theta^{k,l}} P\{\theta^{k,l}|Z^k\} \quad (2)$$

The proposed algorithm is based on the Viterbi Algorithm (VA). The VA is a dynamic programming algorithm for determining the optimal state sequence in a trellis diagram [2].

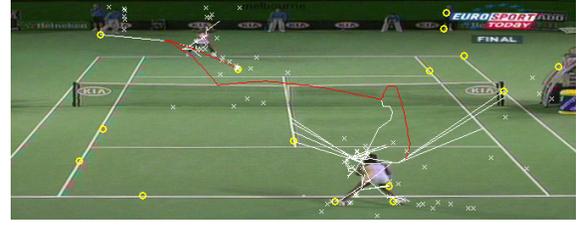


Figure 1. The MAP-VDA in operation. Yellow circles: ball candidates at the current time step. White crosses: predicted ball positions given by the “survived” Kalman filters in the previous time step. Lines: “survived” paths at the current time step. Red line: the path with lowest overall distance so far. White lines: other paths.

The key idea of the VA is, for each node at each level of a trellis, to keep only the history and total “distance” of the “best” path that goes into it. Having obtained ball candidates in all K fields in the sequence, we construct a trellis in such a way that each field in the sequence corresponds to a level in the trellis, and each candidate $c_i(k)$ corresponds to a node $n_i(k)$. Also an extra “dummy” node $n_0(k)$ is introduced at the k^{th} level of the trellis, which corresponds to the case that the ball is not detected in field k . Assuming that nodes $n_{i'}(1)$ and $n_{i''}(K)$ are known to correspond to the true object, and also assuming a distance metric between two adjacent nodes is defined, the VA can then be applied to find the path between $n_{i'}(1)$ and $n_{i''}(K)$ with, for instance, the shortest overall distance.

We define the arc distance between two nodes $n_{i_{k-1,s}}(k-1)$ and $n_{i_{k,l}}(k)$ as:

$$\begin{aligned} d^l[n_{i_{k-1,s}}(k-1), n_{i_{k,l}}(k)] &= -\ln \\ &[P\{Z(k)|\theta_{i_{k,l}}(k), \theta^{k-1,s}, Z^{k-1}\} \times P\{\theta_{i_{k,l}}(k)|\theta^{k-1,s}, Z^{k-1}\}] \end{aligned} \quad (3)$$

From Eq. 1, Eq. 2 and Eq. 3, the shortest path given by the VA corresponds to the sequence of measurements we are looking for, *i.e.*, the one with maximum *a posteriori* probability.

By assuming false candidates are uniformly distributed in the image plane, the likelihood term in Eq. 3 is [1]:

$$\begin{aligned} P\{Z(k)|\theta_{i_{k,l}}(k), \theta^{k-1,s}, Z^{k-1}\} &= \\ \begin{cases} V_k^{-(m_k-1)} P_G^{-1} \mathcal{N}[\mathbf{z}_{i_{k,l}}(k); \hat{\mathbf{z}}^s(k|k-1), \mathbf{S}^s(k)] & i_{k,l} \neq 0 \\ V_k^{-m_k} & i_{k,l} = 0 \end{cases} \end{aligned} \quad (4)$$

and the prior probability term in Eq. 3 is:

$$\begin{aligned} P\{\theta_{i_{k,l}}(k)|\theta^{k-1,s}, Z^{k-1}\} \\ = P\{\theta_{i_{k,l}}(k)|m_k\} = \begin{cases} \frac{1}{m_k} P_D P_G & i_{k,l} \neq 0 \\ 1 - P_D P_G & i_{k,l} = 0 \end{cases} \end{aligned} \quad (5)$$

where m_k is the number of validated candidates in field k , V_k is the volume of the validation region, P_G is the probability that the true measurement falls in the validation region, P_D is the object detection probability, $\hat{\mathbf{z}}^s(k|k-1)$ is the predicted measurement given by the Kalman filter that is associated with the s^{th} sequence of measurements at time $k-1$, $\mathbf{S}^s(k)$ is the corresponding innovation covariance matrix, and $\mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}, \mathbf{P})$ denotes the normal density with argument \mathbf{x} , mean $\bar{\mathbf{x}}$ and covariance \mathbf{P} . After some reorganisation and normalisation, the final definition of arc distance is:

$$d[n_{i_{k-1},s}(k-1), n_{i_{k,l}}(k)] = \begin{cases} -\ln\{\mathcal{N}[\mathbf{z}_{i_{k,l}}(k); \hat{\mathbf{z}}^s(k|k-1), \mathbf{S}^s(k)]\} & i_{k,l} \neq 0 \\ -\ln\{\frac{1-P_D P_G m_k}{P_D V_k}\} & i_{k,l} = 0 \end{cases} \quad (6)$$

Initialisation at $k = 1$:

- Initialise a Kalman filter with candidate $n_{i'}(1)$.

Repeat for $k = 2, \dots, K-1$:

- For each path that terminates at each node $n_{i_{k,l}}(k)$, there is a Kalman filter associated with it. Now calculate $d[n_{i_{k-1},s}(k-1), n_{i_{k,l}}(k)]$ according to Eq. 6.
- For each $n_{i_{k,l}}(k)$, $i_{k,l} \neq 0$, among all the paths that terminate at it, find the one that has the lowest total distance. Keep the history of this path and update the corresponding Kalman filter with $\mathbf{z}_{i_{k,l}}(k)$. All other paths and corresponding Kalman filters are discarded.
- For node $n_0(k)$, if a path that terminates at it contains more than N_u dummy nodes in the recent N_r levels, the path and the corresponding Kalman filter are discarded.

Termination at $k = K$:

- Among all the paths that terminate at $n_{i''}(K)$, find the one that has the lowest total distance. This path corresponds to the sequence of measurements with maximum *a posteriori* probability.

Algorithm 1: MAP-VDA

2.2 Bi-directional MAP-VDA

MAP-VDA works well when the linear Gaussian assumption is satisfied. However, when the ball changes its motion drastically, the tracking can go wrong. Fig. 2 gives an example of how this may happen. In Fig. 2, the ball changes its motion in field 75, and is not detected in field 76 and 77. As a result, the next detected ball position (in field 78) falls outside the validation region of the Kalman filter that is associated with the correct path. In other words, the

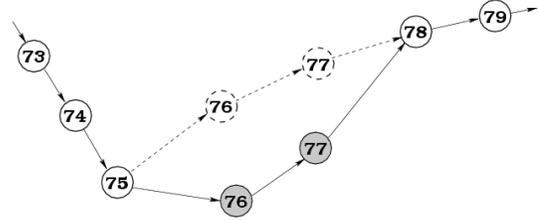


Figure 2. Tracking error caused by abrupt motion change: forward MAP-VDA is “forced” to take a wrong path (solid lines), since the correct path (dashed lines) does not exist in the trellis. White solid circles: the true object; shadowed solid circles: clutter-originated blobs; dashed circles: misdetections (corresponding to the dummy nodes in the trellis).

correct path does not exist in the trellis, and the MAP-VDA algorithm is “forced” to take a wrong path.

If we apply MAP-VDA backwards, however, the story is different. In the reversed time order, the part of the trellis that has wrong path in the forward pass (field 75 to 78 in Fig. 2) now comes “before” the abrupt motion change. As a result, the true candidate in field 75 falls inside the validation region of the Kalman filter that is associated with the correct path, and the backward MAP-VDA gives correct result.

The shortest path given by forward MAP-VDA and that given by backward-VDA are compared. Regions where the forward and backward paths differ are found. In each of these regions the total distances for both sub-paths are compared, and the sub-path with the shorter distance is selected as the final data association result. By combining forward and backward MAP-VDA this way, tracking errors caused by abrupt motion changes of the object is recovered.

3. Experiments

Experiments were carried out on 25 sequences from the 2003 Australian Open tennis tournament women’s final match to track a tennis ball. Each sequence was a play shot starting from a serve. In total the 25 sequences were approximately 5 minutes long, and contained $N = 13099$ in-play fields. The dimension of each field was 268×670 pixels.

We used temporal differencing to extract foreground moving objects. Due to occlusion and motion blur, the ball was extracted as a foreground blob in only $N_d = 12143$ fields. An SVM was then used to classify foreground blobs into ball candidates and non-balls. Features used in the

SVM included shape, colour and position of each blob [8]. By moving the decision boundary of the SVM, a trade-off could be made between true positive rate r_{tp} and false positive rate r_{fp} .

A simple algorithm, which looked for smoothly moving objects, was used to initialise the tracker from both ends, *i.e.*, to identify $n_{i'}(1)$ and $n_{i''}(K)$. For the initialisation, r_{tp} was set to 0.95. The overall detection rate was then $r_d = r_{tp} \times (N_d/N) = 0.88$. The average number of false positives in each field was $\bar{m}_{fp} = 0.3$. Since r_d was high and \bar{m}_{fp} was low, the tracker was correctly initialised in all 25 sequences. To test the performance of the algorithm in heavily cluttered environment, after the initialisation we set $r_{fp} = 1$. Effectively, the blob classifier was not used to reduce false positives. As a result, r_d increased slightly to 0.93, but \bar{m}_{fp} also increased significantly to 11.6.

We compare the proposed bi-directional MAP-VDA algorithm with our previous work in [8], where a particle filter (PF) is combined with a smoothing algorithm to track the tennis ball. The performance of both algorithms, measured by distribution of tracking precision, is shown in Fig. 3. To calculate the distributions, ground truth of the tennis ball positions in all in-play fields was manually labelled. The tracking results were then compared against the ground truth. The tracking precision was defined as the Euclidean distance between the ground truth and the tracked (detected or interpolated) ball position.

It is shown in Fig. 3 that the performance of the proposed algorithm is comparable to that of the PF in [8]. It should be noted that in order for the PF to work, a large number of particles had to be used, which made the PF slow. The average processing time of the PF was 2 fields/second, while for the proposed algorithm, it was approximately 11 fields/second. Furthermore, the PF in [8] required the tennis players to be tracked in order to properly switch between its two dynamic models, while the proposed algorithm did not require this information. Time was measured on a Pentium 4 2.8G computer, and overhead of disk I/O was included.

4. Conclusions

In this paper, we propose a data association algorithm for ball tracking in broadcast sports video. The proposed algorithm seeks the sequence of measurement-to-target associations with maximum *a posteriori* probability. Under the Markov assumption of the object motion, this optimal sequence of associations is obtained efficiently by applying the Viterbi algorithm. We also develop a method for combining forward and backward tracking results, to recover from tracking errors caused by abrupt motion changes of the object. The proposed bi-directional MAP-VDA algorithm was applied to broadcast tennis video to track a tennis ball. Experiments show that the performance of the proposed al-

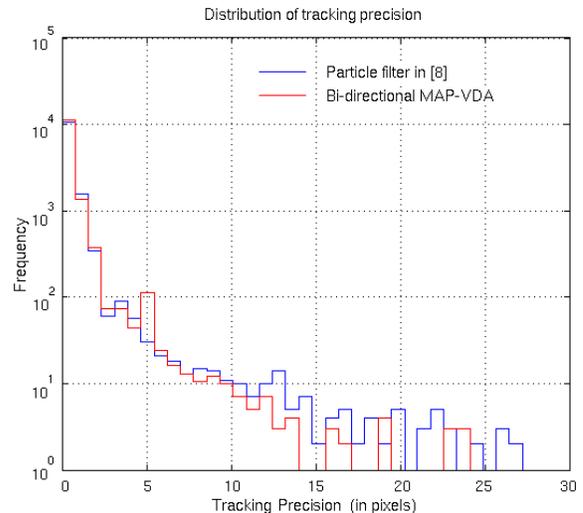


Figure 3. Distribution of tracking precision.

gorithm is comparable to that of a computationally more expensive particle-filter-based algorithm. In the future, we would like to apply the algorithm to other ball games, *e.g.*, soccer, baseball.

References

- [1] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press INC., 1988.
- [2] G. Fomey. The viterbi algorithm. *Proceedings of the IEEE*, pages 268–278, 1973.
- [3] N. Owens, C. Harris, and C. Stennett. Hawk-eye tennis system. In *International Conference on Visual Information Engineering*, 2003.
- [4] G. S. Pingali, Y. Jean, and I. Carlbom. Real time tracking for enhanced tennis broadcasts. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 260–265, 1998.
- [5] G. S. Pingali, A. Opalach, and Y. Jean. Ball tracking and virtual replays for innovative tennis broadcasts. In *IEEE International Conference on Pattern Recognition*, pages 4152–4156, 2000.
- [6] T. Quach and M. Farooq. Maximum likelihood track formation with the viterbi algorithm. In *IEEE Conference on Decision and Control*, pages 271–276, 1994.
- [7] M. Takahashi, T. Misu, M. Tadenuma, and N. Yagi. Real-time ball trajectory visualization using object extraction. In *The 2nd IEE European Conference on Visual Media Production*, pages 60–67, 2005.
- [8] F. Yan, W. Christmas, and J. Kittler. A tennis ball tracking algorithm for automatic annotation of tennis match. In *British Machine Vision Conference*, volume 2, pages 619–628, 2005.
- [9] X. Yu, C. Xu, Q. Tian, and H. Leong. A ball tracking framework for broadcast soccer video. In *International Conference on Multimedia and Expo*, volume 2, pages 273–276, 2003.