OFF-LINE MULTIPLE OBJECT TRACKING USING CANDIDATE SELECTION AND THE VITERBI ALGORITHM

François Pitié, Sid-Ahmed Berrani, Anil Kokaram and Rozenn Dahyot

Electrical & Electronic Engineering Dept.
University of Dublin, Trinity College, Dublin 2. Ireland.
E-mail: fpitie@tcd.ie

ABSTRACT

This paper presents a probabilistic framework for off-line multiple object tracking. At each timestep, a small set of deterministic *candidates* is generated which is guaranteed to contain the correct solution. Tracking an object within video then becomes possible using the Viterbi algorithm. In contrast with particle filter methods where candidates are numerous and random, the proposed algorithm involves a few candidates and results in a deterministic solution. Moreover, we consider here off-line applications where past and future information is exploited. This paper shows that, although basic and very simple, this candidate selection allows the solution of many tracking problems in different real-world applications and offers a good alternative to particle filter methods for off-line applications.

1. INTRODUCTION

Tracking visual objects in image sequences is a key task for a wide range of applications in different domains (traffic surveillance, video summarisation, etc.). It has been extensively studied and many methods have been proposed. Particle filter based methods have become very popular indeed. They are powerful, simple and can handle complex situations in particular multiple objects tracking [1, 2]. They are specially suitable for applications where on-line processing is required. In many such applications past information is used to determine the current position of the tracked object(s).

The First Key Idea in this paper is to acknowledge that in many applications, tracking could be performed off-line. Information retrieval in sport footage [3] and video indexing are typical examples of such applications. In this context, a global analysis of the video can be performed to extract object paths, that is, visual features are first extracted from all the frames and then analysed in a second step. In such a scenario exploiting both the past and future information could lead to useful gains.

The Second Key Idea in this paper is to consider the possibility of generating at each timestep, a candidate set of solutions that is guaranteed to contain at least one solution that is *correct*. In that case a deterministic process can yield the MAP estimate for tracking. This may seem wishful, yet it is worthwhile to consider this alternative route to tracking because such simple scenarios do indeed exist and can arise from realistic problems. Given the difficulties posed by the correct application of particle filters, in particular the problem of degeneracy of particles, it is useful to consider alternative strategies where those are viable. The success of this approach depends entirely on the process for generating candidates, it must be simple, and reliable enough that the candidates always contain the *correct* state. It is interesting to note that Kernel Particle Filter in [4] have introduced the notion of pre-processing as a means of improving particle diversity in the particle filter for a tracking problem. The reader can consider that this paper takes that idea one step further and proposes that if the candidate selection stage is reliable enough (which it can be) sampling can be avoided.

Organisation of the Paper. An overview of the methodology for off-line object tracking is presented in section 2. It is explained how, by defining a suitable candidate selection and a set transition probabilities, tracking an object within the video becomes equivalent to finding the most likely path in the candidate trellis using the Viterbi algorithm [5].

Although basic and very simple the candidate selection process allows the solution of many tracking problems in different real-world applications. It also allows the easy integration of specific rules related to the object motion. The paper presents in sections 3 and 4 two applications that represent domains in which tracking is amenable to this kind of idea.

The first application considered aims at detecting the arms of a child in a psychological assessment exercise. Tracking is used only to take temporal information into account and avoid false detections due to occlusions. It is a sim-

This work has been partly founded by HEA TRIP, Enterprise Ireland MUSEDTV, DYSVIDEO, and EU project MUSCLE FP6-507752 www.muscle-noe.org.

ple application that allows to introduce the framework. The second application is more challenging and concerns player detection and tracking in soccer video footages. Problems of introduction of a new players in the scene, disappearance of a tracked player and occlusions have to be dealt with.

2. OVERVIEW OF THE METHODOLOGY

Consider that $\mathbf{x}_n$ is the random variable corresponding to the object position $\mathbf{x}$ (which may be 1 or 2D depending on the application), where $n \in [1; N]$, and N is the number of frames of the sequence. Bayes theorem states that the *posterior* distribution of the object position throughout the sequence $\mathbf{x}_{1:N}$ can be written as

$$p(\mathbf{x}_{1:N}|\mathbf{y}_{1:N}) \propto p(\mathbf{x}_{1:N})\, p(\mathbf{y}_{1:N}|\mathbf{x}_{1:N}) \qquad (1)$$

where $p(\mathbf{x}_{1:N})$ corresponds to the *prior* on the object positions and $p(\mathbf{y}_{1:N}|\mathbf{x}_{1:N})$ corresponds to the *likelihood* for the object positions given the data model $\mathbf{y}_{1:N}$—which corresponds here to the frames of the sequence.

We assume that the likelihood can be computed independently on each frame $p(\mathbf{y}_{1:N}|\mathbf{x}_{1:N}) = \prod_n p(\mathbf{y}_n|\mathbf{x}_n)$. In general there are a large number of possible states (each pixel location in each image, and in each frame). Reducing the number of states will reduce the computational load. The idea is to propose a limited number of states as *candidates* from some pre-process. One option is to generate these candidates as the peaks of the likelihood $p(\mathbf{y}_n|\mathbf{x}_n)$. The likelihood presenting r peaks is then approximated by the following grid-based distribution:

$$p(\mathbf{y}_n|\mathbf{x}_n) \propto \sum_{i=1}^{r} p(\mathbf{y}_n|\mathbf{x}_n^{(i)})\, \delta(\mathbf{x}_n^{(i)} - \mathbf{x}_n) \qquad (2)$$

The candidates solutions of the tracking follow some *rules* depending on the application (feasible moves, scenarios of occlusions, ...). These rules are encapsulated in the prior function which gives the transition probabilities between candidates. Then from the rules and the candidates we can apply Viterbi [5] to extract the most likely path, which is actually a Maximum a Posteriori estimation. For more than one object, we can iteratively apply Viterbi and remove the corresponding candidates from the set of candidates. The *posterior* of the successive tracks is decreasing and the number of objects to track can be automatically determined by thresholding the *posterior*.

The success of the method depends on the simplicity of the object detector which is performed on the whole picture. A few particle filter trackers [6, 2] propose a similar approach by sampling part of the particles directly from the likelihood. For instance [2] uses Adaboost to detect the entrance of new players.

A somehow similar candidate refinement as been used in Kernel Particle Filter [4]. But here candidates are fully *deterministic* as well the resulting tracking. This implies also that our method requires much less candidates.

3. APPLICATION TO A SIMPLE CASE STUDY

In this section, we show how the framework can be applied in a real application. The application aims at detecting the position of the hands of a child performing a psychological exercise [7] as presented in figure 2.

Candidates are found by projecting the colour skin segmentation [7] of the frames on the horizontal axis and taking the main peaks of the projection as candidates for the hands positions (see figure 2). The likelihood of these candidates is proportional to the value of the projection. The presence of the instructor can generate spurious peaks in the skin colour projection and we select up to the 5 most important peaks.

Transition probabilities are set to prevent large displacements of the hands: $p(x_n|x_{n-1}) \sim \mathcal{N}(0, 3)$

Once the hand candidate positions have been collected, we can apply the Viterbi algorithm to extract one hand trajectory. To track the other hand, it suffices to remove the candidates corresponding to the first track and then to apply again Viterbi on the reduced set of candidates. The figure 2 shows some example of results (see also [8]).

4. APPLICATION TO MULTIPLE OBJECTS TRACKING

This section proposes a more difficult tracking application: the tracking of soccer players [9]. We need to extract candidate positions of the players and to set the rules explaining the dynamic of the players.

4.1. Player Candidate Positions

Playground Extraction. The playground can be efficiently extracted using a colour segmentation of the pitch followed by simple morphological operations to fill in holes and remove spurious detections.

Player Detection. Colour is a relevant feature to characterise players. As shown in figure 3 a colour segmentation will result in a map of blobs corresponding to the players. It is possible then using a Mean Shift procedure to extract the centre of mass of these blobs and locate the player. This method has to be related to the colour histogram based mean-shift techniques used in [10, 11] which are known to be robust. The method is also fast, because the mean shift can be done on downsized pictures, and generates a small

(a) (b) (c)

Fig. 1. Example of player occlusion: the blue player on (b) is fully occluded. The tracking method assigns temporary the position of the red player to the blue one.

set of candidates for each frame (typically less than 20 candidates).

4.2. Set of Rules

From the set of candidate players, we need to explicit in particular how a player can move, how it can appear or disappear from the field of the camera and lastly how it can be occluded by other players.

Player Motion. Even though the frames are not registered, a player displacement of 50 pixels represent a minimum pace of 50km/h and is impossible ($p(\mathbf{x}_n|\mathbf{x}_{n-1}) = 0$).

Player Apparition/Disappearance. Players can only appear and disappear on the borders of the frames. To allow this event at any time of the video, we add two abstract states positions $\mathbf{x}_a$ and $\mathbf{x}_b$. $\mathbf{x}_a$ indicates that the player is not yet visible and $\mathbf{x}_b$ indicates that the player is not any more visible. It is then possible to express transition probabilities. In particular $p(\mathbf{x}_n = \mathbf{x}_a|\mathbf{x}_{n-1} = \mathbf{x}_b) = 0$ (which means that a player cannot appear more than once). We can express also the probability of apparition by $p(\mathbf{x}_n \notin \{\mathbf{x}_a, \mathbf{x}_b\}|\mathbf{x}_{n-1} = \mathbf{x}_a) = g(\mathbf{x}_n)$, where g is a decreasing function of the distance of the candidate to the border of the frame and is equals to 0 when the candidate is 50 pixels away from the frame border.

Occlusions by another Team-Mate. In this case the colour detection will only spot one player, and this single candidate position corresponds to two different tracks. To overcome this problem, each time Viterbi has been run for a player, instead of removing candidates from the pool, the candidate positions are kept but penalised by reducing their likelihood (division by 3). Penalising the previously selected candidates avoid to generate multiple instance of the same track but still allows for temporary overlap of the tracks.

Occlusions by an Opponent. If a player is occluded by an opponent, its colour is also occluded and the player cannot be detected. In this situation we use the candidate positions of the opponent team. Practically we add the candidate positions of the other team to the current set of candidates, but with a much lower likelihood (division by 3). Figure 1 shows the results obtained for such a scenario. The blue player is not found on the middle frame and is temporary assigned to the position of the red player.

Post-Processing Rules. Since the model is based on a Markov Chain of order 1, post processing rules allow for integrating richer features to filter the results. One can decide if a player visible only on a few frames is worth being tracked. One can also set, as mentioned earlier in the paper, a threshold for the *posterior*. If the *posterior* of the tracking is too small, the object is insignificant and the multitracking process stops.

Figure 4 shows some results for the tracking of the blue team and video material is available online at [8]. It is noteworthy that the results are obtained deterministically and can be reproduced identically, whereas with particle filter methods, the results are partly random and will differ slightly in applications to the same footage with the same initial conditions, with a non-zero possibility of outright failure in any given instance.

5. CONCLUSION

In this paper we have shown that in some applications, when the image data is such that the object detection task is quick and robust, 1) random candidate generation in particle filters can be efficiently replaced by a deterministic candidate selection that results in deterministic solutions and 2) for off-line applications the Viterbi algorithm can be applied to exploit the whole available temporal information.

6. REFERENCES

[1] C. Hue, J.-P. Le Cadre, and P. Pérez, "Sequential monte carlo methods for multiple target tracking and data fusion," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 309–325, February 2002.

[2] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *European Conference on Computer Vision (ECCV)*, 2004.

[3] "Special Session on Sports Video Analysis (chaired by I. Sezan and B.Li)," in *IEEE International Conference on Image Processing (ICIP)*, Sep 2003.

[4] C. Chang and R. Ansari, "Kernel particle filter: Iterative sampling for efficient visual tracking," in *IEEE International Conference on Image Processing (ICIP)*, 2003.

[5] Charles W. Therrien, *Decision estimation and classification: an introduction to pattern recognition and related topics*, John Wiley & Sons, Inc., 1989.

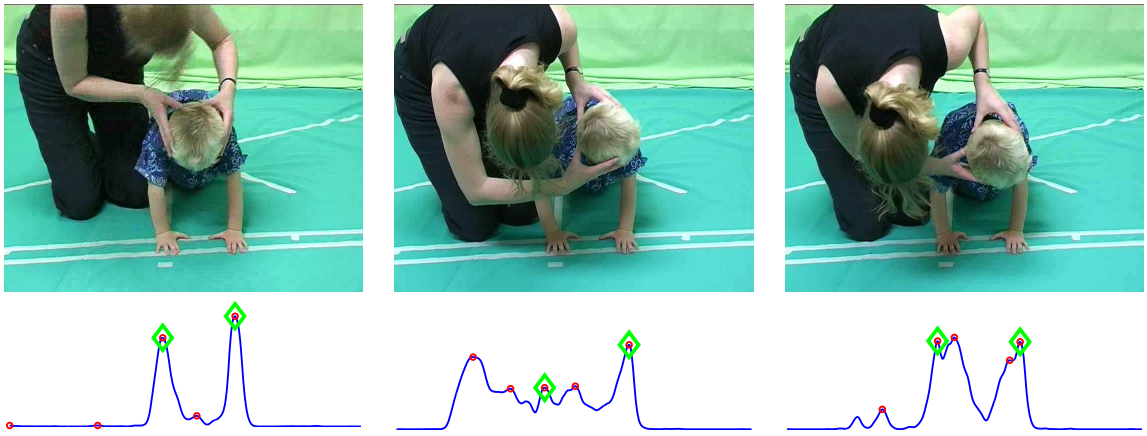

Fig. 2. Detection of the hands positions (green diamonds) of a child performing a psychological exercise [7]. The peaks of the skin colour projection give the candidate positions (red circles).

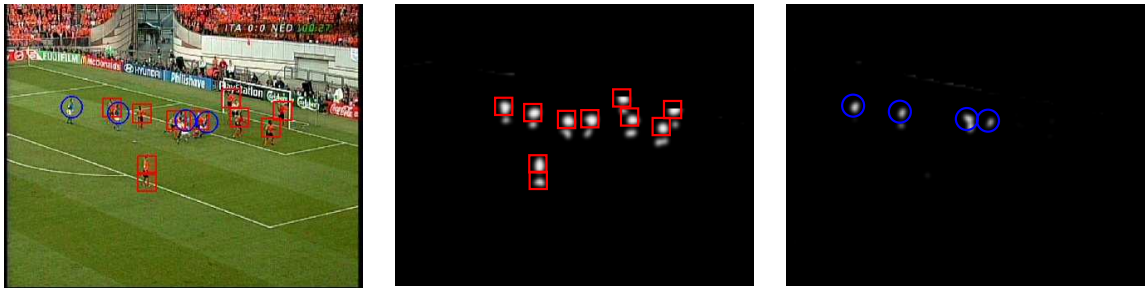

Fig. 3. Example of Player Detection using colour segmentation and MeanShift to find the blobs centres.

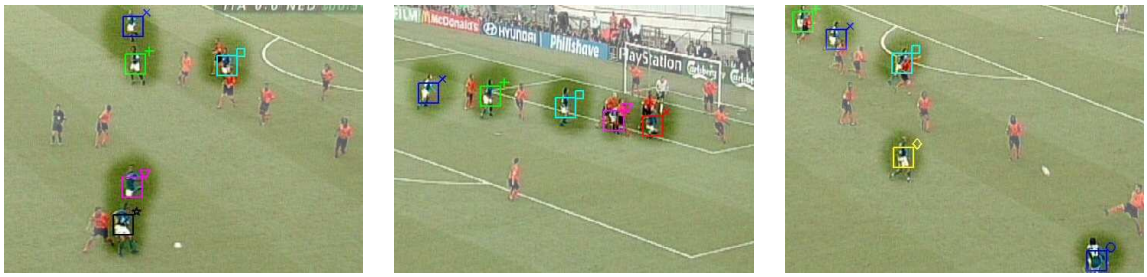

Fig. 4. Tracking in action on the soccer sequence. Only the blue team is tracked, other areas are in lower contrast. Videos are available at [8]. (Image courtesy of Rádio e televisão Portugal)

[6] B. Terwijn, J.M. Porta, and B.J.A. Kröse, "A particle filter to estimate non-markovian states," in *International Conference on Intelligent Autonomous Systems, IAS'04*, 2004.

[7] L. Joyeux, E. Doyle, H. Denman, A. Crawford, A. Kokaram, and R. Fuller, "Content based access for a massive database of human observation video," in *Workshop on Multimedia and Image Retrieval*, October 2004, pp. 46–52.

[8] F. Pitié, "Video material," Website, 2005, `http://www.mee.tcd.ie/~sigmedia/publications/publis/ICIP2005/fpitie/`.

[9] Hyun-Wook OK, Yongduek Seo, and K.S. Hong, "Multiple soccer players tracking by condensation with occlusion alarm probability," in *International Workshop on Statistical Methods for Vision Processing (in conjunction with ECCV)*, 2002.

[10] G. Jaffré and A. Crouzil, "Non-rigid object localization from color model using mean shift," in *IEEE International Conference on Image Processing (ICIP)*, 2003.

[11] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 5, pp. 564–575, 2003.