Visual tracking and auxiliary discrete processes

Patrick Pérez[1] and Jaco Vermaak[2]

[1] IRISA/INRIA, Campus Universitaire de Beaulieu
35042 Rennes Cedex, France
(e-mail: perez@irisa.fr)

[2] Cambridge Univ. Eng. Dpt., Trumpington St.
Cambridge, CB2 1PZ, United Kingdom
(e-mail: jv211@eng.cam.ac.uk)

Abstract. A number of Bayesian tracking models involve auxiliary discrete variables beside the main hidden state of interest. These discrete variables usually follow a Markovian process and interact with the hidden state either via its evolution model or via the observation process, or both. Examples of such auxiliary variables include depth ordering for occlusion handling, switches between different state dynamics, exemplar indices, etc. We consider here a general model that encompasses all these situations, and show how Bayesian filtering can be rigorously conducted in this general setup. The resulting approach facilitates easy re-use of existing tracking algorithms designed in the absence of the auxiliary process. In particular we show how particle filters can be obtained based on sampling only in the original state space instead of sampling in the augmented space, as it is usually done. We finally demonstrate how this framework facilitates solutions to the critical problem of appearance and disappearance of targets, either upon scene entering and exiting, or due to temporary occlusions. This is illustrated in the context of color-based tracking with particle filters.

Keywords: Optimal Bayesian filter, Auxiliary discrete process, Particle filter, Visual tracking, Occlusion, Disappearance, Object detection.

1 Introduction and motivation

Visual tracking involves the detection and recursive localization of objects within video frames. In a number of visual trackers, the state of interest, e.g., size and location of the object, is associated with auxiliary discrete variables. Such variables show up for instance within the state evolution model, e.g., when different types of dynamics can occur (e.g., [North *et al.*, 2000]). More often, such auxiliary variables are introduced in the observation model. It is the case for appearance models based on a set of key views (e.g., [Toyama and Blake, 2001],[Wu *et al.*, 2003]) or silhouettes (e.g., [Gavrila, 2000] [Toyama and Blake, 2001]). Auxiliary variables are also used to handle partial or total occlusions (e.g., [Nguyen *et al.*, 2001]) or mutual occlusions when jointly tracking multiple objects (e.g., [MacCormick and Blake, 1999] [Wu *et al.*, 2003]). Finally, auxiliary variables can be used to assess the presence of tracked objects in the scene (e.g., [Vermaak *et al.*, 2002] [Isard and MacCormick, 2001]). When a Bayesian tracking approach is used with such augmented models, either specific filters are derived based on the detailed form of the model at hand or the optimal filter of the joint model is simply used. In the latter case, a practical implementation might be unnecessarily costly due to the increased dimension of the joint space. Sequential Monte Carlo approximations (SMC) in the joint space are for instance used in [Isard and MacCormick, 2001] [Toyama and Blake, 2001] [Vermaak *et al.*, 2002] [Wu *et al.*, 2003].

The first contribution of this paper is to propose a general and unified framework to easily derive the optimal Bayesian filter for the augmented model based on the one for a model with no (or frozen) auxiliary variables. In practice, this allows the re-use of existing tracking architectures, with a reasonable computational overhead in case the discrete auxiliary variable only takes a small number of values. This approach allows us in particular to introduce a generic SMC architecture that relies on sampling in the main state space only. This is exposed in Section 2.

The problem of appearing and disappearing objects, whether it is upon entering and exiting the scene, or upon getting occluded by another object, is critical in visual tracking. As we mentioned above, the different forms of this problem have already been addressed in the past based on auxiliary hidden processes. The second contribution of this paper is to re-visit these problems using our generic framework. The resulting filters are implemented using the generic SMC architecture proposed in Section 2. To handle occlusions, we introduce in Section 3 a binary visibility process that intervenes in the observation model. In this case, our generic approach allows us to derive a two-fold mixture filter that deal with temporary occlusions. In a similar fashion, we address the problem of "birth" and "death" of objects, which is crucial for multiple-object tracking, by introducing a binary existence process. This process impacts both the state evolution and the data model. The application of our approach leads in this case to a simple filter whose SMC approximation does not need to draw samples for the existence variable.

2 Tracking with an auxiliary process

2.1 Modeling assumptions

For visual tracking, we are interested in recursively estimating the object state $\mathbf{x}_t \in \mathbb{R}^{n_x}$, which specifies the position of the object in the image plane and, possibly, other parameters such as its size and orientation, based on a sequence of observations $\mathbf{y}^t = (\mathbf{y}_1 \cdots \mathbf{y}_t)$. We assume in addition that a discrete auxiliary variable a_t also has to be recursively inferred. This variable takes its values in a set of cardinality M that we will denote by $\{0 \cdots M-1\}$ for convenience.

The complete set of unknowns at time t is thus $\{\mathbf{x}_t, a_t\}$, for which we assume the following Markovian prior

$$p(\mathbf{x}_t, a_t|\mathbf{x}_{t-1}, a_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}, a_t, a_{t-1})p(a_t|a_{t-1}). \tag{1}$$

In other words, the state follows a Markov chain with its kernel parameterized by the current and previous values of the auxiliary variable, and the auxiliary process is a discrete Markov chain. Let $A = (\alpha_{ji})$ be its $M \times M$ transition matrix, with $\alpha_{ji} \doteq p(a_t = i|a_{t-1} = j)$. For brevity, we will also use the notation

$$p_{ji}(\mathbf{x}_t|\mathbf{x}_{t-1}) \doteq p(\mathbf{x}_t|\mathbf{x}_{t-1}, a_t = i, a_{t-1} = j). \tag{2}$$

As for the observation model, we assume in the normal way that the image data at successive instances are independent conditional on the hidden variables, i.e., $p(\mathbf{y}_t|\mathbf{x}_t, a_t, \mathbf{y}^{t-1}) = p(\mathbf{y}_t|\mathbf{x}_t, a_t)$. For notational convenience we will denote

$$p_i(\mathbf{y}_t|\mathbf{x}_t) \doteq p(\mathbf{y}_t|\mathbf{x}_t, a_t = i). \tag{3}$$

2.2 Bayesian filter

For tracking, we are interested in recursively estimating the joint filtering distribution

$$p(\mathbf{x}_t, a_t|\mathbf{y}^t) = p(\mathbf{x}_t|a_t, \mathbf{y}^t)p(a_t|\mathbf{y}^t), \tag{4}$$

from which the marginal filtering distribution can be deduced as

$$p(\mathbf{x}_t|\mathbf{y}^t) = \sum_i p(\mathbf{x}_t, a_t = i|\mathbf{y}^t) = \sum_i p_i(\mathbf{x}_t|\mathbf{y}^t)\xi_{i,t}, \tag{5}$$

where we used the notation

$$p_i(\mathbf{x}_t|\mathbf{y}^t) \doteq p(\mathbf{x}_t|a_t = i, \mathbf{y}^t) \text{ and } \xi_{i,t} \doteq p(a_t = i|\mathbf{y}^t). \tag{6}$$

Similar to our previous notation, we will now use the distribution subscript i to indicate conditioning with respect to the current auxiliary variable set to i, and the distribution subscript ji for conditioning on i and j being the current and previous values of the auxiliary variable.

We will first show how to compute the M conditional state posteriors $p_i(\mathbf{x}_t|\mathbf{y}^t)$. First note that

$$p_i(\mathbf{x}_t|\mathbf{y}^t) = \frac{p_i(\mathbf{x}_t, \mathbf{y}_t|\mathbf{y}^{t-1})}{p_i(\mathbf{y}_t|\mathbf{y}^{t-1})}. \tag{7}$$

The numerator can be expressed as

$$p_i(\mathbf{x}_t, \mathbf{y}_t|\mathbf{y}^{t-1}) = \sum_j p_{ji}(\mathbf{x}_t, \mathbf{y}_t|\mathbf{y}^{t-1})p(a_{t-1} = j|a_t = i, \mathbf{y}^{t-1}), \tag{8}$$

with

$$\begin{aligned}
p_{ji}(\mathbf{x}_t, \mathbf{y}_t|\mathbf{y}^{t-1}) &= p_i(\mathbf{y}_t|\mathbf{x}_t)p_{ji}(\mathbf{x}_t|\mathbf{y}^{t-1}) \\
&= p_i(\mathbf{y}_t|\mathbf{x}_t)\int p_{ji}(\mathbf{x}_t|\mathbf{x}_{t-1})p_j(\mathbf{x}_{t-1}|\mathbf{y}^{t-1})d\mathbf{x}_{t-1}, \quad (9) \\
p(a_{t-1}=j|a_t=i,\mathbf{y}^{t-1}) &\doteq \tilde{\alpha}_{ji,t} \\
&\propto p(a_t=i|a_{t-1}=j,\mathbf{y}^{t-1})p(a_{t-1}=j|\mathbf{y}^{t-1}). \quad (10)
\end{aligned}$$

Based on the conditional independence structure of the model, one can show that the first term on the right hand side is independent of $\mathbf{y}^{t-1}$. We thus obtain, after normalization,

$$\tilde{\alpha}_{ji,t} = \frac{\alpha_{ji}\xi_{j,t-1}}{\sum_k \alpha_{ki}\xi_{k,t-1}}. \tag{11}$$

The predictive likelihood in the denominator of (7) is

$$p_i(\mathbf{y}_t|\mathbf{y}^{t-1}) = \sum_j \tilde{\alpha}_{ji,t}\int p_{ji}(\mathbf{x}_t,\mathbf{y}_t|\mathbf{y}^{t-1})d\mathbf{x}_t. \tag{12}$$

The filtering distribution in (5) is then a mixture of the M conditional filtering distributions, i.e.,

$$p_i(\mathbf{x}_t|\mathbf{y}^t) = \frac{\sum_j \tilde{\alpha}_{ji,t}p_{ji}(\mathbf{x}_t,\mathbf{y}_t|\mathbf{y}^{t-1})}{p_i(\mathbf{y}_t|\mathbf{y}^{t-1})}, \tag{13}$$

each of which is obtained by combining M optimal Bayesian filters to compute (9) and (12).

We still need the marginal posterior of the auxiliary variable, $p(a_t|\mathbf{y}^t)$, to compute the weights $\xi_{i,t}$ in the mixture of (5). We have

$$\xi_{i,t} \propto p_i(\mathbf{y}_t|\mathbf{y}^{t-1})\sum_j p(a_t=i|a_{t-1}=j,\mathbf{y}^{t-1})\xi_{j,t-1}. \tag{14}$$

Since the first factor in the sum is independent of $\mathbf{y}^{t-1}$, we finally obtain, after normalization

$$\xi_{i,t} = \frac{p_i(\mathbf{y}_t|\mathbf{y}^{t-1})\sum_j \alpha_{ji}\xi_{j,t-1}}{\sum_k p_k(\mathbf{y}_t|\mathbf{y}^{t-1})\sum_j \alpha_{jk}\xi_{j,t-1}}. \tag{15}$$

Let us summarize the operations at time t for the generic algorithm:

- **Input**: $p_i(\mathbf{x}_{t-1}|\mathbf{y}^{t-1})$ and $(\xi_{i,t-1})$ for $i = 0 \cdots M-1$.

1. Compute $\tilde{\alpha}_{ji,t}$ as in (11), for $i = 0 \cdots M-1$.
2. Compute distributions $p_{ji}(\mathbf{x}_t,\mathbf{y}_t|\mathbf{y}^{t-1})$ as in (9), for $i, j = 0 \cdots M-1$.
3. Compute distributions $p_i(\mathbf{y}_t|\mathbf{y}^{t-1})$ as in (12), for $i = 0 \cdots M-1$.
4. Compute filtering distributions $p_i(\mathbf{x}_t|\mathbf{y}^t) =$ as in (13), for $i = 0 \cdots M-1$.

5. Compute posterior distribution $(\xi_{i,t})_{i=0\cdots M-1}$ of auxiliary variable as in (15).

- **Output**: distributions $p_i(\mathbf{x}_t|\mathbf{y}^t)$ and weights $\xi_{i,t}$.

At each time step, M^2 "elementary" filtering operations are required (step 2), one per possible occurrence of the pairing (a_t, a_{t-1}). In practice, not all M^2 values may be admissible, in which case the number of elementary filtering operations at each time step is reduced accordingly. As we will see, specificities of the model under consideration might also permit further computational savings.

The framework above is entirely general, both in terms of model ingredients (evolution and observation processes) and in terms of implementation. Regarding the latter, all existing techniques, whether exact or approximate, can be accommodated. If, for example, the filtering distributions $p_i(\mathbf{x}_t|\mathbf{y}^t)$ are to be represented by Gaussian mixtures, the mixtures components can be obtained by the Kalman filter for linear Gaussian models, and by the extended or unscented Kalman filters for non-linear and/or non-Gaussian models. For models of the latter kind it may sometimes be beneficial to adopt a particle representation, and use sequential importance sampling techniques to update the filtering distribution. This is especially true for the highly non-linear and multi-modal models used in visual tracking, hence the success of SMC techniques in the computer vision community. It is this type of implementation that we now consider.

2.3 SMC implementation

For a general SMC implementation, we will consider proposal distributions of the form $q_{ji}(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_t) \doteq q(\mathbf{x}_t|\mathbf{x}_{t-1}, a_t = i, a_{t-1} = j, \mathbf{y}_t)$. Based on these proposals, different SMC architectures can be designed to approximate the generic algorithm of the previous section. We propose here an architecture that is based on systematic resampling. Assuming that each conditional posterior distribution $p_i(\mathbf{x}_{t-1}|\mathbf{y}^{t-1})$ at time $t-1$ is approximated by a set $(\mathbf{s}^{(n)}_{i,t-1})_{n=1\cdots N}$ of N equally weighted particles, we simply replace steps 2, 3 and 4 in the generic algorithm by:

2. For $j = 0 \cdots M-1$, for $i = 0 \cdots M-1$
   2a. Sample N particles $\tilde{\mathbf{s}}^{(n)}_{ji,t}\mathrm{sim} q_{ji}(\mathbf{x}_t|\mathbf{s}^{(n)}_{j,t-1}, \mathbf{y}_t)$.
   2b. Compute the *normalized* predictive weights

$$\pi^{(n)}_{ji,t} \propto \frac{p_{ji}(\tilde{\mathbf{s}}^{(n)}_{ji,t}|\mathbf{s}^{(n)}_{j,t-1})}{q_{ji}(\tilde{\mathbf{s}}^{(n)}_{ji,t}|\mathbf{s}^{(n)}_{j,t-1}), \mathbf{y}_t} \text{ with } \sum_n \pi^{(n)}_{ji,t} = 1. \tag{16}$$

3. Approximate the M predictive data likelihoods by

$$p_i(\mathbf{y}_t|\mathbf{y}^{t-1}) \approx \sum_j \sum_n w^{(n)}_{ji,t}, \tag{17}$$

where, for $i, j = 0 \cdots M-1$,

$$w_{ji,t}^{(n)} \doteq \tilde{\alpha}_{ji,t} p_i(\mathbf{y}_t | \tilde{\mathbf{s}}_{ji,t}^{(n)}) \pi_{ji,t}^{(n)}. \tag{18}$$

4. For $i = 0 \cdots M-1$, draw N particles $\mathbf{s}_{i,t}^{(n)}$ with replacement from the weighted set $(\tilde{\mathbf{s}}_{ji,t}^{(n)}, p_i(\mathbf{y}_t | \mathbf{y}^{t-1})^{-1} w_{ji,t}^{(n)})_{j,n}$ of $M \times N$ particles.

Steps 1 and 5 remain unchanged. At each instant t, posterior expectations can be approximated using the final particle sets. In particular,

$$\mathbb{E}[\mathbf{x}_t | a_t = i, \mathbf{y}^t] \approx \hat{\mathbf{x}}_{i,t} \doteq \frac{1}{N} \sum_n \mathbf{s}_{i,t}^{(n)}, \ \mathbb{E}[\mathbf{x}_t | \mathbf{y}^t] \approx \hat{\mathbf{x}}_t \doteq \sum_i \xi_{i,t} \hat{\mathbf{x}}_{i,t}. \tag{19}$$

If the proposal distribution does not depend on $a_t = i$, then step 2a can be performed M times instead of M^2 times, providing particles sets $(\tilde{\mathbf{s}}_{j,t}^{(n)})_n$ to be used in place of $(\tilde{\mathbf{s}}_{ji,t}^{(n)})_n$ in the remainder of the algorithm.

3 Appearance and disappearance

Most tracking algorithms assume the number of objects of interest to be constant in the sequence. However, in most cases objects of interest enter and exit the scene at arbitrary times. In addition, they can also disappear temporarily behind other occluding objects. In the latter case of occlusion, tracking should be continued blindly in the hope of locking back onto the objects when they re-appear. An object entering or exiting the scene should in contrast result in initiating or terminating tracking, respectively. In any case, these appearance and disappearance events, whether they are temporary or definitive, are themselves uncertain events. The associated concepts of "existence" and "visibility" should thus be treated jointly with the other unknowns within a probabilistic framework that can account for all the expected ambiguities. Exploiting the generic approach presented in the previous section, we propose to achieve this using two auxiliary binary processes. Although these two processes can be used jointly, we introduce them separately for the sake of clarity.

3.1 Visibility process

Explicit introduction of an occlusion process within the Bayesian tracking framework was proposed in [MacCormick and Blake, 1999] and [Wu *et al.*, 2003]. Both works, however, rely on specific modeling assumption (contour-based tracking in the former, luminance exemplars in the latter), and specific implementations (particle filter with partitioned importance sampling in the former vanilla bootstrap particle filter in the latter). In contrast, our approach relies on generic modeling assumptions and is independent of a specific implementation strategy, so that existing tracking architectures can be re-used. The occlusion modeling we propose can thus be used in conjunction with any Bayesian visual tracking technique, based for instance on the Kalman filter or one of its variants. In addition, using it within the SMC architecture of Section 2 allows restriction of the sampling to the object state space only.

Considering here only the case of complete occlusion, we introduce a binary visibility variable v_t that indicates whether the object is visible ($v_t = 1$) or not ($v_t = 0$) in the image at time t. The Markov chain prior on this binary variable is completely defined by the occlusion and desocclusion probabilities, α_{10} and α_{01}. The state evolution model is independent of the visibility variable, i.e.,

$$p_{ji}(\mathbf{x}_t|\mathbf{x}_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}). \tag{20}$$

Two data models,

$$p(\mathbf{y}_t|\mathbf{x}_t, v_t = 0) = p_0(\mathbf{y}_t) \text{ and } p(\mathbf{y}_t|\mathbf{x}_t, v_t = 1) = p_1(\mathbf{y}_t|\mathbf{x}_t), \tag{21}$$

will have to be specified, depending on whether the object of interest is visible in the image or not. In the former case, the likelihood is independent of the state value. Since our experiments are conducted in the context of color-based tracking we consider a simple observation model related to the more complex ones proposed in [Isard and MacCormick, 2001] and [Vermaak *et al.*, 2002]. Pixel-wise location independent background and foreground models, g_0 and g_1, respectively, are specified over the selected color space. Assuming conditional independence of color measures over a sub-grid S of pixels, we obtain

$$p_0(\mathbf{y}_t) = \prod_{s \in S} g_0(\mathbf{y}_{s,t}) \text{ and } p_1(\mathbf{y}_t|\mathbf{x}_t) = \prod_{s \in R(\mathbf{x}_t)} g_1(\mathbf{y}_{s,t}) \prod_{s \in \bar{R}(\mathbf{x}_t)} g_0(\mathbf{y}_{s,t}), \tag{22}$$

where $R(\mathbf{x}_t)$ is the image region associated with an object parameterized by the state $\mathbf{x}_t$, and $\mathbf{y}_{s,t}$ is the color at pixel s in frame t.

For this dynamic model, the SMC architecture of Section 2 can be simplified. Indeed, the independence of the state evolution with respect to the auxiliary variables allows step 2a to be performed only M times, and suggests the use of a unique proposal. A simple and classical choice is to take the state dynamics (20) as the proposal [Isard and Blake, 1996]. We will adopt this approach here, while bearing in mind that any data-based proposal, including the optimal one [Doucet *et al.*, 2000] in the rare cases that it is accessible, can be used in our generic framework.

Fig. 1 shows results obtained on a sequence where a walking person is successfully tracked despite a succession of severe and total occlusions caused by trees in the foreground. The tracking is initialized manually on the red top of the person. The initialization also provides the reference foreground model g_1, defined as a $5 \times 5 \times 5$ joint histogram in the RGB color space. The histogram for the reference background model g_0 is also obtained in the first frame based on the image complement of the initial selection. The unknown state $\mathbf{x}_t$ comprises the position in the image plane ($n_x = 2$) and its dynamics (20) is taken to be a random walk with independent Gaussian noise with variance 10^2 on each component. The parameters of the Markov chain on the visibility process are $\alpha_{01} = 0.8$ and $\alpha_{10} = 0.1$, and its initial distribution is given by $p(v_0 = 1) = 0.8$. We use $N = 200$ particles for the SMC implementation. The main quantities of interest are the marginal filtering distributions (5), which inform on the localization of the object of interest regardless of whether it is visible or not. We display the MC approximations of the state expectations $\hat{\mathbf{x}}_t$ relative to these distributions in Fig. 1. The algorithm also recursively estimates the marginal visibility posterior $p(v_t = 1|\mathbf{y}^t)$. The time evolution of this quantity for the pedestrian sequence is plotted in Fig. 2. It correctly drops to zero for each complete occlusion of the tracked person.

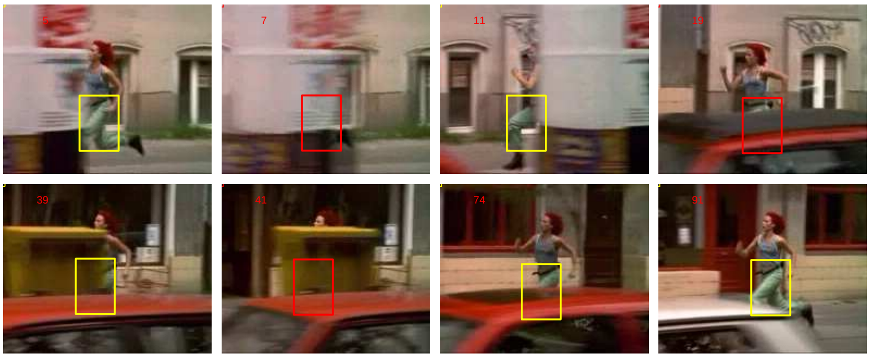

Fig. 1. Tracking under occlusions. The color-based tracker is initialized on the trousers of Lola (from movie "Run, Lola, run") who runs in the street. The rapid succession of partial, large or complete occlusions caused by cars, poles and mailbox is successfully handled thanks to the explicit modeling of visibility changes. In each of the displayed frames, the box corresponds to $\hat{\mathbf{x}}_t$ and its color is changed from yellow to red when $\xi_{1,t}$ drops below 0.5.

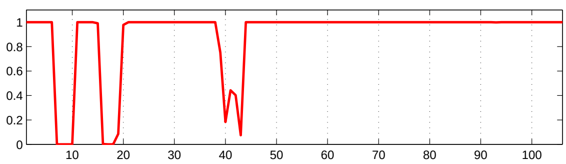

Fig. 2. Posterior visibility probability, $\xi_{1,t} = p(v_t = 1|\mathbf{y}^t)$, plotted against time for the example in Fig. 1. Occlusions and desocclusions make respectively the visibility probability drop, possibly down to zero, and increase back to unity.

3.2 Existence process

Using a Markovian binary variable to indicate presence in the scene is proposed in [Vermaak *et al.*, 2002] to determine in a probabilistic fashion the beginning and end of the track for a single object. We adopt the same model here. However, sequential Monte Carlo is the only inference mechanism considered in [Vermaak *et al.*, 2002], and it is conducted in the augmented state space. By comparison, our generic framework can be easily used with any Bayesian filtering technique and its SMC version implies sampling only in the object state space.

Following [Vermaak *et al.*, 2002], we introduce a binary existence variable e_t that indicates whether the object of interest is present ($e_t = 1$) or not ($e_t = 0$) in the scene at time t. The Markov chain prior on this binary variable is completely defined by the death and birth probabilities, α_{10} and α_{01}. Conditional on the existence variables the state dynamics is specified by

$$p_{00}(\mathbf{x}_t|\mathbf{x}_{t-1}) = p_{10}(\mathbf{x}_t|\mathbf{x}_{t-1}) = \delta_{\mathbf{u}}(\mathbf{x}_t) \tag{23}$$

$$p_{01}(\mathbf{x}_t|\mathbf{x}_{t-1}) = p_{\text{init}}(\mathbf{x}_t) \tag{24}$$

$$p_{11}(\mathbf{x}_t|\mathbf{x}_{t-1}) = p_{\text{dyn}}(\mathbf{x}_t|\mathbf{x}_{t-1}), \tag{25}$$

where $\mathbf{u}$ is the consuming state that corresponds to the object not existing, p_{init} is the initial state distribution, and p_{dyn} is the object dynamic model. From the data model point of view, the existence process is similar to the visibility process.

Due to the component (23) of the evolution model, non-existence $e_t = 0$ deterministically forces $\mathbf{x}_t$ into fictitious state $\mathbf{u}$. This is carried over in the posterior model, yielding

$$p_0(\mathbf{x}_t|\mathbf{y}^t) = \delta_{\mathbf{u}}(\mathbf{x}_t). \tag{26}$$

As a consequence, the algorithm only needs to recursively estimate the conditional filtering distribution for the case of the object existing, i.e., $p_1(\mathbf{x}_t|\mathbf{y}^t)$. Thus, within the SMC framework, only two proposal distributions, q_{01} and q_{11}, are required, instead of four. As in the previous section, we only consider the simple case where these distributions coincide with their counterparts in the evolution model.

In the following experiment, the observation model is defined as in the previous section. Yet again the state comprises the object location in the image plane, and in the state evolution model (24)-(25), p_{init} and p_{dyn} are respectively chosen as the uniform distribution over positions in the image plane and a random walk with independent Gaussian noise. The variance of the noise is 15^2 for each component for the car race sequence in Fig. 3. Also, the state distribution at time $t = 0$ coincides with p_{init}. Hence, contrary to the previous experiment, the tracker is not initialized manually at the beginning of the sequence (the reference foreground model is picked on an arbitrary red

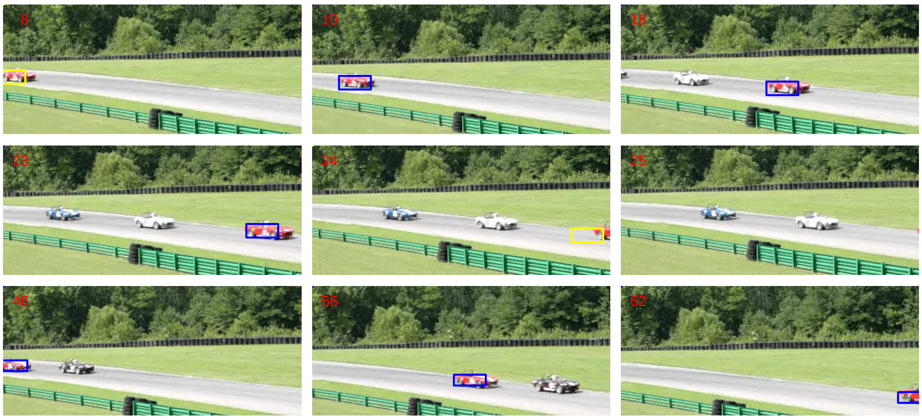

Fig. 3. Detection and tracking. A reference color model is initialized beforehand on one instance of a red car. The algorithm then successfully detects red cars that enter the scene, tracks them as long as they remain in view, and finally determines automatically when they disappear. In each of the displayed frames $\hat{\mathbf{x}}_{1,t}$ is displayed, provided that $\xi_{1,t}$ exceeds 0.2 (in blue if it is greater than 0.8 and in yellow otherwise).

car in a different part of the video). For this experiment, the death and birth probabilities are respectively set to $\alpha_{01} = 0.1$ and $\alpha_{10} = 0.1$, and the initial existence distribution is given by $p(e_0 = 1) = 0.1$. Finally, $N = 50$ particles were sufficient to detect the entrance and exit of red cars in the field of view and to track them while present in the scene. Entrance and exit events are clearly identified by the variations in the posterior existence probability $\xi_{1,t}$, as shown in Fig. 4. In this example, a single tracker successively locks on to different cars, each one appearing in the image after the previous one has been successfully detected and tracked until disappearance. In practice, distinction between different tracked objects would be necessary, especially if they are likely to be present simultaneously in the image. In this context, the information carried by the existence probabilities would facilitate the design of a mechanism that effectively initiates different trackers for each "detected" object and subsequently discards each tracker whose associated existence probability $\xi_{1,t}$ falls below a threshold.

4 Conclusion

In this paper we introduced a generic Bayesian filtering tool to perform tracking in the presence of a certain class of discrete auxiliary processes. The approach places no restriction on the ingredients of the evolution and observation models and on the selected type of filter (Kalman filter and its variants, particle filters). Hence the proposed framework allows re-use of existing architectures on a variety of tracking problems where the introduction

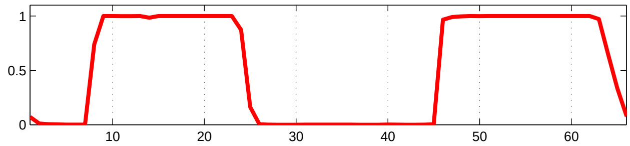

Fig. 4. Posterior existence probability, $\xi_{1,t} = p(e_t = 1|\mathbf{y}^t)$, against time for the example in Fig. 3. When the object of interest enters the scene the existence probability quickly ramps up to one, and falls back down to zero when it exits the field of view.

of auxiliary discrete variables is useful. We demonstrated in particular how the technique can be applied in visual tracking to handle occlusions and object appearance/disappearance via visibility and existence binary processes. Our generic frameworkwould now allow the combination of these two binary processes within a single tracking setup.

References

[Doucet *et al.*, 2000]A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[Gavrila, 2000]D. Gavrila. Pedestrian detection from a moving vehicle. In *Proc. Europ. Conf. Computer Vision*, Dublin, Ireland, June 2000.

[Isard and Blake, 1996]M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. Europ. Conf. Computer Vision*, pages I:343–356, 1996.

[Isard and MacCormick, 2001]M. Isard and J. MacCormick. BraMBLe: a Bayesian multiple-blob tracker. In *Proc. Int. Conf. Computer Vision*, pages II: 34–41, Vancouver, Canada, July 2001.

[MacCormick and Blake, 1999]J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *Proc. Int. Conf. Computer Vision*, pages 572–578, 1999.

[Nguyen *et al.*, 2001]H.T. Nguyen, M. Worring, and R. van den Boomgaard. Occlusion robust adaptive template tracking. In *Proc. Int. Conf. Computer Vision*, pages I: 678–683, Vancouver, Canada, July 2001.

[North *et al.*, 2000]B. North, A. Blake, M. Isard, and J. Rittscher. Learning and classification of complex dynamics. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(9):1016–1034, 2000.

[Toyama and Blake, 2001]K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *Proc. Int. Conf. Computer Vision*, pages II: 50–57, Vancouver, Canada, July 2001.

[Vermaak *et al.*, 2002]J. Vermaak, P. Pérez, M. Gangnet, and A. Blake. Towards improved observation models for visual tracking: selective adaptation. In *Proc. Europ. Conf. Computer Vision*, pages I: 645–660, Copenhagen, Denmark, May 2002.

[Wu *et al.*, 2003]Y. Wu, T. Yu, and G. Hua. Tracking appearances with occlusions. In *Proc. Conf. Comp. Vision Pattern Rec.*, Madison, Wisconsin, June 2003.