Hybrid Joint–Separable Multibody Tracking

Oswald Lanz*
ITC-irst
Via Sommarive 18, 38050 Povo, Italy
lanz@itc.it

Roberto Manduchi
University of California at Santa Cruz
Santa Cruz, 1156 High Street
manduchi@soe.ucsc.edu

Abstract

Statistical models for tracking different moving bodies must be able to reason about occlusions in order to be effective. Representing the joint statistics across different bodies is computationally hard, since the size of the representation grows exponentially with the number of bodies being tracked. Separable tracking, with one tracker per body, cannot deal with occlusions effectively. We propose a new model, dubbed Hybrid Joint-Separable (HJS), that uses a representation size that grows linearly with the number of bodies, and a computational complexity that grows quadratically. This model can reason explicitly about occlusions. We describe a particle filter implementation of this model, and present promising experimental results.

1. Introduction

Visual tracking of multiple moving targets is a challenging problem. Independent tracking of individual bodies fails in the presence of occlusions, where the disappearance of a target cannot be explained if not in relationship with the other targets. On the other hand, describing the dynamics of the different bodies with a joint model requires a representation size and computational cost that grow exponentially with the number of bodies.

We propose a new approach to recursive Bayes tracking that can describe occlusions explicitly, and yet has an approachable complexity. In particular, the representation size of the whole system grows linearly with the number of tracked bodies, K, while the complexity at each upgrade grows quadratically with K. Our strategy is based on an hybrid between separable and joint tracking models. More precisely, we represent the posterior probability (belief) of the joint state using a separable (independent component) model. At each update, a joint likelihood model (which takes occlusions into account) is used to describe the correction to the projected belief determined on the scene appearance. This update produces a non–separable posterior distribution, which is then "marginalized" into a set of K different distributions. We show in this paper that this update sequence can be implemented efficiently using a particle–based representation of the beliefs for the different targets. Although suboptimal. this strategy has shown excellent results in experiments involving multiple occluding bodies.

This article is organized as follows. After a brief review of previous work in the field, we introduce the Hybrid Joint–Separable model in Sec. 2, Sec. 3 describes the basic theory of HJS multibody tracking, while Sec. 4 discuss the implementation of HJS tracking using particle filtering, focusing first on the case of occlusions on a single line of sight, and then extending the algorithm to the general case of full–image tracking. Sec. 5 presents tracking experiments on two sequences with persons moving in a room, demonstrating the power of HJS tracking even in difficult cases with full occlusions. Sec. 6 has the conclusions.

1.1. Previous Work

A probabilistic exclusion principle for tracking was introduced in [7]. By preventing a single pixel from being independently associated to different objects, a robust form for multibody observation probability density was found. The object state is enhanced with a discrete dimension that discriminates between foreground and background hypotheses, thus allowing for occlusion modeling in a principled way. However, this approach is restricted to specific contour-based measurement types which in general do not convey information to distinguish between different objects with similar shape. Tracking using an abstraction to object–level and configuration–level behavior was proposed in [9]. In this work, independent single–object hypotheses are reviewed using heuristics based on blob coverage and compactness. This approach conceals the nature of the tracked probability density, making it difficult to obtain a rigorous probabilistic interpretation. In partially occluded situations,

*O. Lanz has been partly funded by Provincia Autonoma di Trento under Project PEACH: Personalized Experience of Active Cultural Heritage.

feature-based tracking can still be applied with success to points that are suitably selected at each frame [4]. A major drawback of feature–based methods is their focus on local features, while neglecting important higher–level information such as target shape and texture. In addition, an object that becomes completely occluded becomes lost, even if the occlusion lasts for only a few frames. A two–step recursive Bayesian estimation approach for a multiview setup is presented in [8]. This algorithm tracks objects located in the intersections of 2–D visual angles, which are extracted from silhouettes obtained from image segmentation in the different cameras. Occlusion hypotheses are generated and tested using a branch–and–merge strategy. To avoid the combinatorial explosion arising from recursively testing all possible hypotheses, only hypotheses with posterior probability above a certain threshold are kept. In [3], a strategy for solving the track split, merge and overlap problem as arising in segmentation–based approaches during occlusions is proposed. While easily accommodated in real–time, sudden changes in the background are fatal for this kind of methods. Also, appearance model adaptation is prone to learn the occluder's appearance if covered for a period of time.

Compared with these previous approaches, our proposed algorithm has the advantage of: (1) Using descriptive and robust features (color histograms), (2) Relying on sound probabilistic modeling, and (3) Reasoning explicitly about occlusions, while maintaining an approachable computational cost.

2. Hybrid Joint–Separable Models

We will denote by $\mathbf{x}_t = (x_t^1, x_t^2, \ldots, x_t^K)$ the joint state vector at time t (where time assumes only discrete values). The K components of $\mathbf{x}_t$ (which may themselves be vectors) represent the states of the K bodies being tracked. The observation at time t is indicated by z_t; the sequence of observations until time t is denoted by $z_{1:t}$. The pdf of the state at time $t = 0$, $p(\mathbf{x}_0)$, is assumed known. The evolution of the state sequence is described by a Markov process of order one:

$$\mathbf{x}_t = f_t\left(\mathbf{x}_{t-1}, \mathbf{v}_{t-1}\right) \tag{1}$$

where $\mathbf{v}_{t-1}$ is an i.i.d. noise process sequence. The observation z_t depends on $\mathbf{x}_t$ as by:

$$z_t = g_t\left(\mathbf{x}_t, w_t\right) \tag{2}$$

where w_t is an i.i.d. noise process sequence. Our goal is to estimate $p\left(\mathbf{x}_t|z_{1:t}\right)$, the posterior distribution (belief) of $\mathbf{x}_t$ given the observation sequence $z_{1:t}$. Using Bayes' theorem and the total probability theorem, the Chapman–Kolmogorov recursion is found [1] (see Fig. 1(a)):

$$\begin{aligned} & p\left(x_t|z_{1:t}\right) \\ \propto\ & p\left(z_t|\mathbf{x}_t\right) \int p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right) p\left(\mathbf{x}_{t-1}|z_{1:t-1}\right)\, d\mathbf{x}_{t-1} \\ =\ & p\left(z_t|\mathbf{x}_t\right) E_{t-1}\left[p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)\right] \end{aligned} \tag{3}$$

where $E_{t-1}[\cdot]$ represents expectation with respect to the previously found posterior distribution $p\left(\mathbf{x}_{t-1}|z_{1:t-1}\right)$. In other words, $E_{t-1}\left[p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)\right]$ projects the belief of $\mathbf{x}$ at time $t-1$ into the belief at time t before observing z_t:

$$E_{t-1}\left[p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)\right] = p\left(x_t|z_{1:t-1}\right) \tag{4}$$

Exact or approximated solutions to this recursion can be found using Kalman filtering, grid–based methods, or particle filtering [1]. Unfortunately, representing the joint posterior distribution $p\left(\mathbf{x}_t|z_{1:t}\right)$ may be unwieldy. The size of the representation, as well as the cost of computing the recursion in (3), grows exponentially with the number K of bodies being tracked. A simple solution would be to represent and estimate the evolution of the bodies independently. Formally, a separable model is defined by the following two hypotheses:

Separability Hypothesis 1:

$$p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right) = \prod_i p\left(x_t^i|x_{t-1}^i\right) \tag{5}$$

Separability Hypothesis 2:

$$p\left(z_t|\mathbf{x}_t\right) \propto \prod_i p\left(z_t|x_t^i\right) \tag{6}$$

It is easy to show that, when the two separability hypotheses above are satisfied, and if $p\left(\mathbf{x}_0\right)$ is separable (i.e., it factorizes into $\prod_i p\left(x_0^i\right)$), then the posterior distribution $p\left(\mathbf{x}_t|z_{1:t}\right)$ is separable:

$$p\left(\mathbf{x}_t|z_{1:t}\right) = \prod_i p\left(x_t^i|z_{1:t}\right) \tag{7}$$

In this case, K trackers can be implemented independently, one for each body (see Fig. 1(b)):

$$p\left(x_t^i|z_{1:t}\right) = p\left(z_t|x_t^i\right) E_{t-1}^i\left[p\left(x_t^i|x_{t-1}^i\right)\right] \tag{8}$$

where $E_{t-1}^i\left[\cdot\right]$ is the expectation with respect to $p\left(x_{t-1}^i|z_{1:t-1}\right)$. Both storage requirements and computational complexity to implement the recursion in (8) grow linearly with the number K of bodies.

How acceptable are the separable model hypotheses? Hypothesis 1 states that bodies move independently of each other. We argue that this is an acceptable assumption, with the following caveats. First, it cannot model the case of people interacting and congregating. Second, it implicitly assumes that bodies can compenetrate (or that they have very small size). The real problems with the separable model,

however, are associated with Hypothesis 2, which states that different bodies contribute independently to the observation. This is clearly not true in the case of occlusions: if a body is occluded, its contribution to the observation is null. This has been formalized, for example, as a probabilistic exclusion principle for contour-based tracking [7]. Experience has shown that occlusions are a major cause of failure for tracking systems. Hence, Hypothesis 2 is simply not acceptable in practice.

We propose a new model, dubbed *Hybrid Joint–Separable (HJS)*, for describing the relationship between the system's dynamics and the observations. The HJS model allows one to recursively estimate the state with a computational complexity that grows quadratically (rather than exponentially) with the number of bodies, K. The representation size grows linearly with K. The basic idea is to represent posterior distributions via the outer product of their marginals:

$$p\left(\mathbf{x}_t|z_{1:t}\right) \approx \bar{p}\left(\mathbf{x}_t|z_{1:t}\right) = \prod_i p\left(x_t^i|z_{1:t}\right) \tag{9}$$

where the marginal $p\left(x_t^i|z_{1:t}\right)$ is defined by:

$$p\left(x_t^i|z_{1:t}\right) = \int_{\mathbf{x}^{\neg i}} p\left(\mathbf{x}_t|z_{1:t}\right)\, d\mathbf{x}^{\neg i} \tag{10}$$

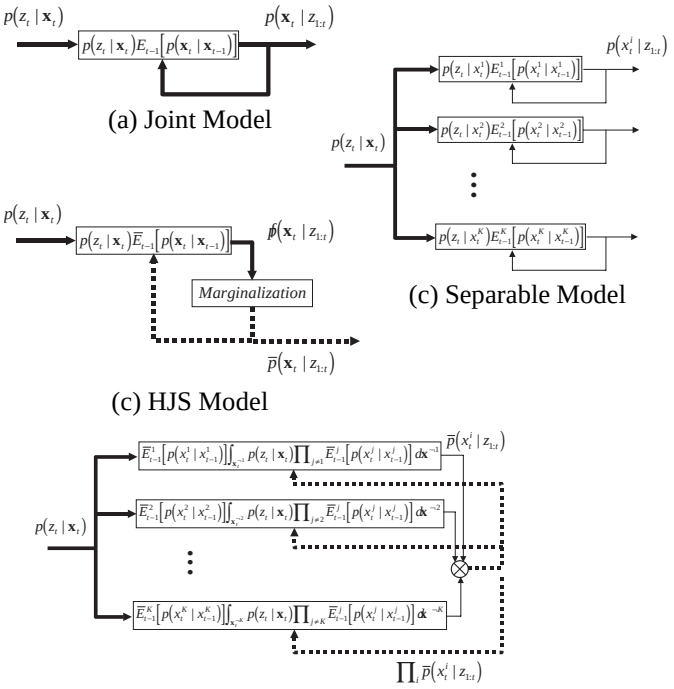
where $\mathbf{x}^{\neg i}$ represent the vector $\mathbf{x}$ with the i–th component removed. The HJS model defines the following update recursion (see Fig. 1(c)):

$$\begin{aligned} \tilde{p}\left(\mathbf{x}_t|z_{1:t}\right) &= p\left(z_t|\mathbf{x}_t\right) \bar{E}_{t-1}\left[p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)\right] \\ \bar{p}\left(\mathbf{x}_t|z_{1:t}\right) &= \prod_i \tilde{p}\left(x_t^i|z_{1:t}\right) \end{aligned} \tag{11}$$

where $\bar{E}_{t-1}\left[\cdot\right]$ represents expectation with respect to $\bar{p}\left(\mathbf{x}_{t-1}|z_{1:t-1}\right)$. Note that we are using the joint form of the conditional likelihood $p\left(z_t|\mathbf{x}_t\right)$, and *not* its separable version (6). Since $p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)$ is assumed to be separable (Hypothesis 1 above), and $\bar{p}\left(\mathbf{x}_{t-1}|z_{1:t-1}\right)$ is separable by definition, it is possible to express (11) as a parallel of systems (see Fig. 1(d)):

$$\begin{aligned} \bar{p}\left(x_t^i|z_{1:t}\right) &= \bar{E}_{t-1}^i\left[p\left(x_t^i|x_{t-1}^i\right)\right] \cdot \\ &\cdot \int p\left(z_t|\mathbf{x}_t\right) \prod_{j\neq i} \bar{E}_{t-1}^j\left[p\left(x_t^j|x_{t-1}^j\right)\right]\, d\mathbf{x}^{\neg i} \\ &= \bar{p}\left(x_t^i|z_{1:t-1}^i\right) \int_{\mathbf{x}^{\neg i}} p\left(z_t|\mathbf{x}_t\right) \prod_{j\neq i} \bar{p}\left(x_t^j|z_{1:t-1}^j\right) d\mathbf{x}^{\neg i} \end{aligned} \tag{12}$$

Note that if the conditional likelihood $p\left(z_t|\mathbf{x}_t\right)$ is separable as in assumption (6), then (12) becomes identical to (8). We will show in Sec. 3 that, for the particular form that $p\left(z_t|\mathbf{x}_t\right)$ takes in the case of occlusion, the complexity of approximating the integral in (12) is linear in K, and therefore the overall complexity of state update is quadratic in K.



(d) HJS Model (Parallel Implementation)

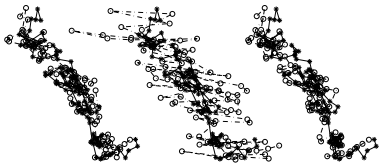
Figure 1. Schematic representation of the models described in Sec. 2. Bold lines represent joint distributions. Dotted lines represent joint distributions expressed as product of marginals. Thin lines represent marginal distributions.

2.1. An Example: HJS Kalman Tracking

The HJS approximation can be applied to virtually any tracking algorithm. Sec. 3 describes its use with particle filtering tracking of multiple bodies. To illustrate the performance of HJS tracking in a simpler context, we present here an example of application with Kalman filtering. Consider the following linear stochastic dynamic system:

$$\begin{aligned} \mathbf{x}_t &= \mathbf{x}_{t-1} + \mathbf{v}_{t-1} \\ z_t &= H\mathbf{x}_t + w_t \end{aligned} \tag{13}$$

where $\mathbf{x}_t$ and z_t are 2–D random variables, $H = \begin{pmatrix} 1 & 5 \\ 0 & 1 \end{pmatrix}$, and $\mathbf{v}_{t-1}$ and w_t are i.i.d. noise process sequences with covariance $\Sigma_v = \Sigma_w = I_2$ (I_2 being the 2×2 identity matrix). Thus, $\mathbf{x}_t$ evolves as an isotropic random walk, which is then skewed by the effect of the observation matrix H. This system satisfies Hypothesis 1 (5) but not Hypothesis 2 (6). A short sequence of a random walk realization of the state $\mathbf{x}$ is shown in Fig. 2(a) (stars), together with the optimal estimation obtained by



(a) (b) (c)

Figure 2. An instance of random walk of the state $\mathbf{x}_t$ for the system (13) (stars), together with the estimated trajectories (circles) using Kalman filtering based on (a) the joint model, (b) the separable model, and (c) the HJS model.

Kalman filtering (circles). Averaged over 10,000 samples, the quadratic tracking error was equal to 2.20. This empirical value is consistent with the asymptotic error covariance, $P_\infty = \begin{pmatrix} 4.55 & -0.87 \\ 0.87 & 0.20 \end{pmatrix}$, found by solving the Riccati equation associated with (13).

A separable approximate model for this system is obtained by rewriting the system equations as follows:

$$\begin{array}{rcl} \hat{\mathbf{x}}_t & = & \hat{\mathbf{x}}_{t-1} + \hat{\mathbf{v}}_{t-1} \\ z_t & = & \hat{\mathbf{x}}_t + w_t \end{array} \tag{14}$$

where $\hat{\mathbf{x}}_t = H\mathbf{x}_t$ and $\hat{\mathbf{v}}_{t-1}$ is an i.i.d. noise sequence with covariance $\hat{Q} = HH^T = \begin{pmatrix} 26 & 5 \\ 5 & 1 \end{pmatrix}$. The separable version of this system is obtained by assuming that the components of $\hat{\mathbf{v}}_{t-1}$ are incorrelated, i.e., that $\hat{Q} = \begin{pmatrix} 26 & 0 \\ 0 & 1 \end{pmatrix}$. Fig. 2(b) shows the estimation of the state trajectory using this separable model for the same sequence as in the previous test. It is clear from the figure that the separable tracker has suboptimal performance; the quadratic error over the same test sequence was equal to 4.03. We repeated the tracking experiment with the HJS model. According to Fig. 1(c), HJS simply marginalizes the joint posterior distribution $p\left(\mathbf{x}_t|z_{1:t}\right)$. In the case of linear Gaussian models (like the one of this example), this operation is equivalent to diagonalizing the posterior error covariance estimated at each iteration. The resulting estimated trajectory, shown in Fig. 2(c), is much closer to the actual one than in the separable case. Indeed, the quadratic tracking error is equal to 2.41, only slightly higher than in the optimal case. Whereas there is probably little computational advantage in using the HJS approximation for Kalman filtering, we'll show in the next section that, for richer models that can describe complex distributions, HSJ can dramatically reduce the computation load and representation size involved in the tracking process.

3. HJS Visual Multibody Tracking

The goal of visual tracking is to monitor the spatial position of one or more moving bodies in the scene. When several cameras are available, one can hope to take direct 3–D measurements by triangulation. With single camera tracking, weak depth information can be inferred by geometric constraints (e.g., by detecting the position of the tracked person's feet on the calibrated ground plane), by measuring foreshortening, or by reasoning on occlusions. We will assume that the state vector for each body being tracked (x_t^i) contains explicit spatial information. For example, x_t^i could represent the body's 3–D position with respect to a fixed coordinate system, together with its velocity. The observation z_t is the projection of the body figure(s) onto the camera plane. If only one body is present in the scene, the conditional likelihood $p(z_t|x_t^i)$ can be defined based on geometric and appearance models. For example, if the camera is calibrated with respect to the chosen world coordinate system, one may compute the position on the image plane of the projection of the body center, together with its expected appearance (defined, for example, by an histogram model). The likelihood $p(z_t|x_t^i)$ could then be defined based on a suitable distance between the predicted image and the actual image. When several bodies are present in the scene, occlusions will occur, meaning that if two bodies are lined up in front of a camera, the closer one will cover view of the other one. In order to express the recursion (12) for the case of multiple occluding bodies, in a way that is amenable to implementation by a particle filter, we consider the simple case of a single line of sight. This could be interpreted as the likelihood relative to a single pixel. We'll show in Sec. 4.3 how to extend this model to the general case of bodies covering possibly large image areas. For simplicity's sake, in this treatment we will assume that each x_t^i simply contains the distance of the i–th body to the camera. Thus, if $x_t^i < x_t^j$, the i–the body is closer to the camera than the j–th body at time t, where both bodies are along the line of sight. In addition, $x_t^i < \mathbf{x}_t^{\neg i}$ means that the i–th body is closer to the camera than any other body, while $x_t^i > \mathbf{x}_t^{\neg i}$ means that the i–th body is not the closest one to the camera. It is clear that the value of the observation, z_t, given the positions of the bodies, is a function only of the closest one. In other words, for a given i:

$$p\left(z_t|\mathbf{x}_t\right) = \begin{cases} p\left(z_t|x_t^i\right) & \text{if } x_t^i < \mathbf{x}_t^{\neg i} \\ p\left(z_t|\mathbf{x}_t^{\neg i}\right) & \text{otherwise} \end{cases} \tag{15}$$

where $p\left(z_t|x_t^i\right)$ represents the likelihood of the observation of the unoccluded i–th body. In our derivation, we will make use of the following notation. Given an integer k between 1 and K, and any subset V of $\Re^K$, let S^k be the subset of points $\mathbf{x}$ in V such that $x^k < \mathbf{x}^{\neg k}$. It is easy to see that, modulo a set of measure zero, the set of the S^k form a

partition of V. Hence, we can re–write (12) as follows:

$$\begin{aligned}
&\bar{p}\left(x_t^i|z_{1:t}\right) = \bar{p}\left(x_t^i|z_{1:t-1}^i\right) \\
&\cdot \Big[\int_{x_t^i<\mathbf{x}_t^{\neg i}} p\left(z_t|\mathbf{x}_t\right) \prod_{j\neq i} \bar{p}\left(x_t^j|z_{1:t-1}^j\right) d\mathbf{x}^{\neg i} \\
&+ \int_{x_t^i>\mathbf{x}_t^{\neg i}} p\left(z_t|\mathbf{x}_t\right) \prod_{j\neq i} \bar{p}\left(x_t^j|z_{1:t-1}^j\right) d\mathbf{x}^{\neg i}\Big]
\end{aligned} \tag{16}$$

$$\begin{aligned}
&= \bar{p}\left(x_t^i|z_{1:t-1}^i\right) \Big[p\left(z_t|x_t^i\right) \int_{x_t^i<\mathbf{x}_t^{\neg i}} \prod_{j\neq i} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ d\mathbf{x}_t^{\neg i} \\
&+ \textstyle\sum_{k\neq i} \int_{S^k} p\left(z_t|\mathbf{x}_t^{\neg i}\right) \prod_{j\neq i} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ d\mathbf{x}_t^{\neg i}\Big] \\
&= \bar{p}\left(x_t^i|z_{1:t-1}^i\right) \Big[p\left(z_t|x_t^i\right) \prod_{j\neq i} \int_{x_t^i<x_t^j} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ dx_t^j \\
&+ \textstyle\sum_{k\neq i} \int_{S^k} p\left(z_t|x_t^k\right) \bar{p}\left(x_t^k|z_{1:t-1}\right) \\
&\cdot \textstyle\prod_{j\neq i,k} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ d\mathbf{x}_t^{\neg i}\Big]
\end{aligned}$$

$$\begin{aligned}
&= \bar{p}\left(x_t^i|z_{1:t-1}^i\right) \\
&\cdot \Big[p\left(z_t|x_t^i\right) \textstyle\prod_{j\neq i} \left(1 - \int_{x_t^j<x_t^i} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ dx_t^j\right) \\
&+ \textstyle\sum_{k\neq i} \int_{x_t^k<x_t^i} p\left(z_t|x_t^k\right) \bar{p}\left(x_t^k|z_{1:t-1}\right) \\
&\cdot \textstyle\prod_{j\neq i,k} \left(1 - \int_{x_t^j<x_t^k} \bar{p}\left(x_t^j|z_{1:t-1}\right)\ dx_t^j\right)\ dx_t^k\Big]
\end{aligned}$$

where in this case V is the subset of the points $\mathbf{x}^{\neg i}$ such that $x^i > \mathbf{x}^{\neg i}$. Eq. (16) reveals an important property about the posterior distribution $\bar{p}(x_t^i|z_{1:t})$: its value at x_t^i only depends on the values of $\bar{p}(x_t^j|z_{1:t-1})$, with $j \neq i$, for $x_t^j < x_t^i$. This property is key to the efficient implementation of HJS particle filtering described in the next section.

4. HJS Particle Filtering

4.1. Particle Filter: Background

The idea underlying particle filters is to maintain a compressed representation of the estimated belief density via a set of representative sample states, the particles. In Monte Carlo sampling, these samples are chosen i.i.d. distributed according to the density $p(x)$ that they should represent. If the density $p(x)$ is difficult to sample, one can sample from some other feasible importance density $g(x)$, correcting the introduced sampling bias by sample weighting. Formally, if x_n are the samples obtained by sampling from $g(x)$, the importance sampling approximation of an arbitrary density function $p(x)$ can be described by:

$$P(\mathcal{A}) = \int_{\mathcal{A}} p(x)\ dx \approx \sum_n \pi_n\ \delta_{\mathcal{A}}(x_n),\ \pi_n = \frac{p(x_n)}{g(x_n)} \tag{17}$$

The weighted particle set $\{\langle x_n, \pi_n\rangle\}$ can therefore be used to represent $p(x)$. The expectation over $p(x)$ of any given function of interest $f(x)$ can be estimated by:

$$E\big[f(x)\big] = \int f(x)p(x)\ dx \approx \frac{1}{\pi}\sum_n f(x_n)\pi_n \tag{18}$$

with the normalization factor π given by the sum of weights π_i. The estimated state is often taken to be the mean, which is obtained setting $f(x) = x$ in the above relation. Given a weighted particle representation $\{\langle x_n, \pi_n\rangle\}$ for the belief at time $t-1$, the recursion in (3) becomes (modulo a normalization factor):

$$p\left(x_t|z_{1:t}\right) \approx\ p(z_t|x_t) \sum_n p(x_t|x_n)\pi_n. \tag{19}$$

A common choice for the importance density is the mixture density derived from the dynamical model:

$$g(x_t) = \sum_n p(x_t|x_n)\pi_n. \tag{20}$$

At each iteration t, a new set of representative samples $\{\bar{x}_n\}$ is sampled from $g(x)$. Then, the observation z_t is used to compute the new importance weights $\bar{\pi}_n = p(z_t|\bar{x}_n)\pi_n$. Due to the usually diffusive behavior of the dynamical model, the weight distribution becomes more and more skewed with each iteration. To avoid this degeneracy problem, particles are periodically resampled according to their weights . This is the basic version of the particle filter, also known as CONDENSATION algorithm. For a more detailed introduction see [5, 6, 1].

4.2. Implementation of HJS Particle Filtering

Algorithm 1 shows an efficient particle filter implementation of the HJS recursion in (16). Each object's belief is represented via N weighted particles, $\{\langle x_n^i, \pi_n^i\rangle\}$. The particles are independently projected form time $t-1$ to time t by sampling N times from the importance mixture density in (20)

$$\{\bar{x}_n^i\} \overset{\text{i.i.d.}}{\sim} \sum_n p(x_t^i|x_n^i)\pi_n^i \approx \bar{E}_{t-1}^i\big[p(x_t^i|x_{t-1}^i)\big]\,. \tag{21}$$

This can be done in two steps. First, weighted resampling (taking $O(N)$ time [7]), is performed at each iteration, by sampling from the discrete distribution $\{\pi_n^i\}$. Each sample represents one mixture component $p(x_t^i|x_n^i)$ in (21). Then, for each sample, a new particle is sampled from the selected mixture components. A procedure for sampling from $p(x_t^i|x_n^i)$ can usually be derived directly from the state dynamics model [1]. These new particles are initially assigned identical weights.

Finally, weights are assigned to the particles based on the observation z_t. This operation *cannot* be performed independently for each particle. Algorithm 1 describes the weight allocation procedure, which is based on (16). This procedure visits particles from the closest one to the camera to the farthest one. For each body i, two buffers, b_{fg}^i and b_{bg}^i, are used to incrementally compute the weight factors

Algorithm 1 HJS update in a particle filter

input:

$\{\langle x_n^i, \pi_n^i \rangle\}$ are particle sets representing $p(x_{t-1}^i | z_{1:t-1})$

sampling:

foreach object index i **do**

resample N particle indexes i.i.d. according to $\{\pi_n^i\}$

foreach selected particle index p **do**

sample new particle $\bar{x}_n^i$ i.i.d. according to $p(x_t^i | x_p^i)$

$\{\langle \bar{x}_n^i, 1 \rangle\}$ are particle sets representing $\bar{E}_{t-1}^i[p(x_t^i | x_{t-1}^i)]$

weighting:

order $\{\bar{x}_n^i\}$ according to camera distance: $\{\bar{x}_p, i_p\}$

initialize buffers $\{b_{\text{fg}}^i = N, b_{\text{bg}}^i = 0\}$

for $p = 1, \ldots, P$ **do**

compute single–body likelihood $q_p = p(z_t | \bar{x}_p)$

$\bar{\pi}_p = q_p \prod_{j \neq i_p} b_{\text{fg}}^j + \sum_{j \neq i_p} b_{\text{bg}}^j$

$b_{\text{fg}}^{i_p} = b_{\text{fg}}^{i_p} - 1$

foreach object index $j \neq i_p$ **do**

$b_{\text{bg}}^j = b_{\text{bg}}^j + q_p \prod_{k \neq j, i_p} b_{\text{fg}}^k$

$\{\langle \bar{x}_n^i, \bar{\pi}_n^i \rangle\}$ are particle sets representing $p(x_t^i | z_{1:t})$

and offsets of the single–body likelihoods. b_{fg}^i accounts for single–body likelihood:

$$b_{\text{fg}}^i \approx 1 - \int_{x_t^j < x_t^i} p\left(x_t^j | z_{1:t-1}\right) \, dx_t^j \tag{22}$$

while b_{bg}^i accounts for occlusion evidence:

$$\begin{aligned} b_{\text{bg}}^i \approx & \sum_{k \neq i} \int_{x_t^k < x_t^i} p\left(z_t | x_t^k\right) p\left(x_t^k | z_{1:t-1}\right) \\ & \cdot \prod_{j \neq i,k} \left(1 - \int_{x_t^j < x_t^i} p\left(x_t^j | z_{1:t-1}\right) \, dx_t^j\right) \, dx_t^k, \end{aligned} \tag{23}$$

Note that, for each particle p, the buffer b_{fg}^i is simply the overall number of particles N, minus the number of particles for the i–th body already visited before p. Hence, when a particle p associated the i–th body is visited, only the buffer b_{fg}^i needs to be updated. At the same time, the buffers b_{bg}^j for $j \neq i$ need to be updated. For each visited particle, the single–body likelihood is computed first. Then, the new particle weight is calculated according to Equation (16), by utilizing b_{bg}^i and b_{fg}^i. Finally, the contribution of the visited particle is accumulated in the affected buffers. The complexity for calculating particle weights is $O(NK^2)$, while the ordering can be done in $O(NK \log NK)$, this last term becoming negligible for reasonably sized particle sets.

4.3. Occlusion Reasoning: The General Case

The theory developed in the previous sections was based on the assumption that bodies are aligned along a single line of sight. If partial occlusions are neglected, it is easy to extend this theory to the general case of bodies figures with finite extent in the image. A straightforward approach would be to regard the entire camera's field of view as a single cone-of-sight. This approximation, however, is unacceptable: the closest target to the image would be assumed to occlude all other targets, even if its appearance is of limited extent. Instead, we propose to subdivide the camera's field of view adaptively into several cones–of–sight. Within one cone–of–sight, states are assumed to be aligned, with the ones in the front fully occluding the ones in the back.

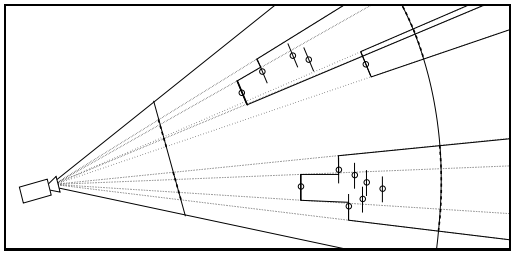


Figure 3. The compound occlusion volumes generated by a target with a bimodal belief. Dotted lines show their slices at a hypothetic particle depth and their image projections.

Observed from a given viewpoint, each opaque object generates a cone–shaped occlusion volume, formed by the 3–D space points that it covers. However, being its location only estimated in form of a set of representative samples, the inferred occlusion structure is no longer described by a single cone. Instead, each object's occlusion volume is represented by a compound of cone segments, each element generated by a different hypothesis of the same target. Fig. 3 shows an example. When a particle is analyzed for its weight, the corresponding cone–of–sight is generated as the union of selected occlusion volumes belonging to the other bodies.

The occlusion volumes associated with different bodies can be generated and compounded incrementally, in parallel to weight calculation. For each particle, a coarse projected shape model is assumed. When visiting a new particle, its projected shape is examined for significant intersection with the other objects' compounded occlusion volumes. Those with significant intersection are considered as the particle's cone–of–sight. The particle's weight is then computed based on Algorithm 1, instantiated within the identified cone-of-sight. If its weight is sufficiently high, the particle's occlusion cone is then accumulated into the compound volume of the object it belongs to.

We used an efficient representation for the compound occlusion volumes. In order to identify the cone–of–sight, or HJS tracker instance, associated with a particle, only the

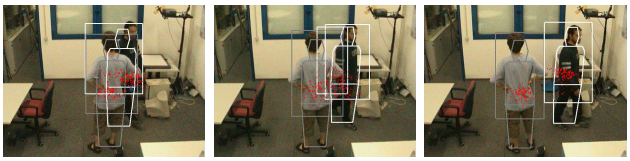


Figure 5. Tree subsequent frames showing ephemerality of phantom hypotheses. Rectangles show projected bounding boxes of the different targets' compound occlusion volumes.

spherical slices of the other objects' occlusion volumes, located at the particle depth, need to be considered. In addition, its intersection with the particle's shape can be performed on the image plane. Thus, occlusion volumes can be efficiently represented by the image–plane projection of the identified cone segment. An efficient implementation based on Algorithm 1 computes occlusion volumes as a sequence of projected slices. In particular, bounding boxes are suitable representation which have been used in the experiments.

5. Experiments

Experiments have been carried out on video sequences captured by a single camera in a small office. Up to three persons were moving randomly in the camera's field of view at the same time, leading to significant partial or even complete occlusions.

Target states were defined by the bodies' positions on a calibrated horizontal reference plane. More precisely, each state x_t^i contains the 2–D position and velocity of the i–th target.

Each body's appearance was modeled by two pre–acquired rough RGB color histograms, one for the head and one for the torso. The single–body log–likelihood of any given target was represented by the sum of the L_1 distances of its model histogram pair to the corresponding candidate histogram pair. Candidate histograms were determined as follows. A pair of 3–D points identifying the hypothetic position of the backbone were obtained from the particle state and the target's known physical height. A calibrated camera model computed the projections of such points on the image plane, around which a coarse planar human silhouette shape was fitted. Candidate color histograms for head and torso were extracted from the data within such silhouettes, and used for likelihood calculation. Cones–of–sight were computed from the silhouette bounding boxes in order to further reduce their representation complexity.

HJS tracking of the persons in the room was implemented as discussed in the previous section. In order to satisfy real–time constraints, only 150 particles per target were used. Note that, since the posterior distribution $p\left(\mathbf{x}_t|z_{1:t}\right)$ is represented by its marginals, a limited number of particles, proportional to the number of bodies K, can be used.

The three targets could enter the room only from one door, at different but known times. Their initial beliefs were sampled from a Gaussian distribution located at the entrance. A linear dynamical model with Gaussian noise was assumed. Thus, at each time, particles were sampled from a mixture of Gaussians whose components are centered at linearly propagated particles, thus implementing (21).

Fig. 4 compares the performance of a separable tracker (top row) and HJS tracking (bottom row) for the same sequence. The separable tracker loses one target immediately after the first short-term occlusion, and can not recover from its failure. This is due in part to the severe behavior of the L_1 norm, which rejects hypotheses much sooner than other types of histogram distances do, e.g. Bhattacharrya distance [2]. It was the intent of the authors to use a discriminant distance, in order to show the improved robustness of HJS. Indeed, HJS succeeds at correctly tracking targets in the same sequence.

Fig. 6 shows a very challenging sequence with two persons entirely covered by a third one. Being not occluded, the target in the foreground is reliably tracked. Thus, the generated cone–of–sight covers closely its true occlusion volume, wherein particles of the other objects remain supported even though not observed. Even objects with similar appearance (such as the two individuals with similar gray clothing in the sequence maintain their identity during occlusions. A separable tracker would be prone to merge the two into the visible, best–fitting target. It should also be noted that depth is tracked to an extent that is sufficient for the HJS to reason explicitly about occlusions even with a single camera.

Fig. 5 shows another interesting situation. Right after a short but complete occlusion, the belief of the occluded target shows a bimodal behavior, as can be noticed from the envelope outlines of the different occlusion volume components. This is due to the likelihood contribution of the occluder, which allocates some phantom particles inside its estimated occlusion volume. However, these false hypotheses are released as soon as the target becomes fully visible and some particles acquire high likelihood, as shown in the right image by means of disappearing occlusion volume components.

6. Conclusions

A novel probabilistic framework for multiple object tracking has been presented. The HJS model has been proposed as a mathematically rigorous methodology for Recursive Bayesian filtering with a reduced representation size.

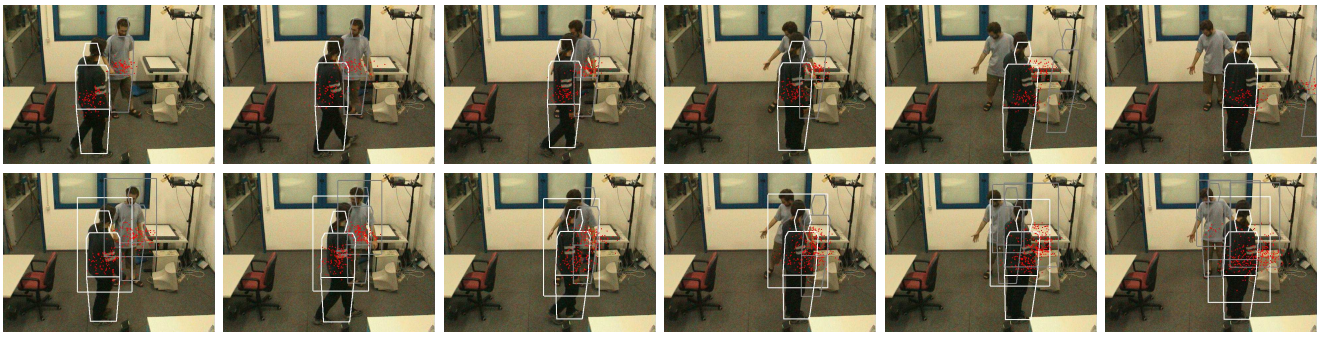


Figure 4. A typical sequence where tracking with different instances of separable trackers fails (top row). The same sequence is successfully tracked by the HJS particle filter (bottom row).

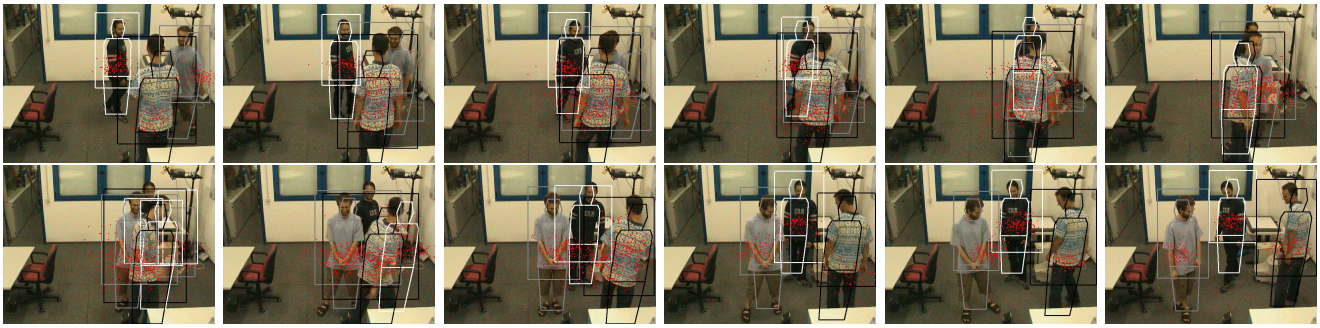


Figure 6. A challenging sequence involving three targets that are momentarily aligned along one line of sight. Although not very accurate during occlusions, the tracker never loses the targets.

The occlusion process has been modeled to derive an algorithm that scales quadratically with the number of objects. Although formalized for a single line–of–sight, this algorithm can be instantiated for several, adaptively selected cones–of–sight, thus accommodating robust image–based likelihoods.

Due to the discrete formulation of the occlusion relation used in this work (in the sense that an object is either completely occluded or fully visible), the derived tracking algorithm has limited accuracy during partial occlusions. The authors are currently investigating the possibility of accommodating occlusion reasoning at pixel–level combined with robust appearance features. The hope is to increase tracking accuracy during partial occlusions while preserving the algorithm's efficiency.

References

[1] S. Arulampalam, A. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on–line non–linear/non–gaussian bayesian tracking. *IEEE Trans. Signal Processing*, 2(50):174–188, February 2002.

[2] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean-shift. In *Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, 2000.

[3] R. Cucchiara, C. Grana, G. Tardini, and R. Vezzani. Probabilistic people tracking for occlusion handling. In *Int. Conf. Pattern Recognition (ICPR)*, 2004.

[4] S. Dockstader and A. Tekalp. Multiple camera tracking of interacting and occluded human motion. *Proc. of the IEEE*, 89(10):1441–1455, 2001.

[5] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[6] M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 1998.

[7] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *Int. Conf. Computer Vision (ICCV)*, 1999.

[8] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, 2004.

[9] H. Tao, H. S. Sawhney, and R. Kumar. A sampling algorithm for detecting and tracking multiple objects. In *Vision Algorithms*, 1999.