Approximate Bayesian Multibody Tracking

Oswald Lanz

O. Lanz is with the Istituto Trentino di Cultura (ITC-irst), Via Sommarive 18, 38050 Povo di Trento, Italy.

E-mail: lanz@itc.it

Abstract

Visual tracking of multiple targets is a challenging problem, especially when efficiency is an issue. Occlusions, if not properly handled, are a major source of failure. Solutions supporting principled occlusion reasoning have been proposed but are yet unpractical for online applications. This paper presents a new solution which effectively manages the trade-off between reliable modeling and computational efficiency. The *Hybrid Joint-Separable (HJS) filter* is derived from a joint Bayesian formulation of the problem, and shown to be efficient while optimal in terms of compact belief representation. Computational efficiency is achieved by employing a *Markov random field approximation* to joint dynamics and an *incremental algorithm* for posterior update with an appearance likelihood that implements a physically-based model of the occlusion process. A particle filter implementation is proposed which achieves accurate tracking during partial occlusions, while in case of complete occlusion tracking hypotheses are bound to estimated occlusion volumes. Experiments show that the proposed algorithm is efficient, robust and able to resolve long term occlusions between targets with identical appearance.

Index Terms

Computer Vision, Tracking, Occlusion, Approximate Inference, Bayes Filter, Particle Filter

I. INTRODUCTION

Visual tracking of multiple moving targets is a challenging problem. Independent tracking of individual bodies is a simple solution but fails in the presence of occlusions, where the disappearance of a target cannot be explained but in relationship with the other targets. On the other hand, principled modeling of the occlusion process is possible when considering the joint configuration of all involved targets, and enables a single tracker in charge of estimating the joint dynamics of the different bodies to interpret images correctly during occlusion. This solution, however, requires a representation size that grows exponentially with the number of bodies, thus leading to an estimation algorithm whose computational complexity grows exponentially as well.

To get a closer understanding of the problem, one may consider the following question:

What distinguishes the problem of tracking the position and the velocity of a single target from the one of tracking the position only of two different targets?

Although both tasks can be formalized as a joint estimation problem, in the first case physical constraints impose a strong correlation of position and velocity, while in the second case the

two components, the locations of the different objects, may depend only weakly from each other, if at all. Their measurements, however, may still be strongly correlated due to occlusions. Based on these considerations, this article presents two main contributions which allow to solve the multitarget tracking problem at an affordable computational cost. The first contribution generalizes and anchors the work presented in [1], providing a novel Bayesian framework tailored to sequential estimation problems of weakly coupled signals, like the ones describing trajectories of different targets to which we will refer from now on. Involved joint prior and posterior distributions (or *beliefs*) are represented by the outer product of single target components, while updates are carried out using a joint dynamical and likelihood model. These updates produce non-separable distributions which are mapped into single target spaces by a projection that guarantees minimal information loss. The key feature of the resulting model, dubbed *Hybrid Joint-Separable (HJS)*, is its economical representation size that scales linearly with the number of targets. The second contribution is the presentation of an occlusion robust multitarget appearance likelihood and an associated algorithm for an efficient update within the HJS model. The likelihood model is derived according to image formation principles and implements occlusion reasoning at pixel level, thus overcoming the limitation of a discrete formulation of the occlusion relation in [1]. The complexity of the HJS posterior update is quadratic in the number of tracked objects and linear in representation size of single target estimates. In addition, joint dynamics is described by a *forward Markov random field (MRF) model*, which corrects independently propagated single target distributions by biasing them with a MRF. Temporal update then relies on Belief Propagation (BP) over this graph structure which preserves the method's efficiency.

A. Related work

Several attempts have been made to find manageable solutions to multibody tracking while guaranteeing robustness. Partitioned sampling [2] alleviates the high computational load associated with the joint approach by decomposing the joint configuration space into 1-body subspaces and performing updates separately and consecutively on them. This method takes a hard decision about occlusions, in the sense that occlusions are resolved based on distributions rather than on single configurations. Thus, the situation in which distributions are not layered w.r.t. a camera view (e.g. a target located at the trough of a bimodal belief belonging to a second object) cannot be handled correctly. The work in [3] reviews that of [2] by proposing a more

consistent particle filter for tracking many targets with the same exclusion principle. The key idea is to jointly process hypotheses of different targets if and only if occlusion exists. Such dependencies are described through *subordination links* between particles belonging to different objects. These links are assigned using a probabilistic propagation scheme rather than derived from imaging principles. Tracking using an abstraction to object-level and configuration-level behavior was proposed in [4] where independent single object hypotheses are validated 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. A Bayesian approach for a multiview setup is presented in [5]. This algorithm tracks objects located at the intersections of visual angles measured at the floor plane, 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 combinatorial explosion inherent in the formulation. This method represents occlusion structures in a 1D projective space (the floor plane), with the drawback that objects generate occlusion volumes that are not limited to their physical height; extension to a fully 2D formulation might require complex descriptions of occlusion geometry. In [6], only the Probability Hypothesis Density (PHD) of the multitarget posterior, i.e. its first moment, is propagated and its particle filter formulation becomes practical. This approach maintains the joint structure of the problem, even if at a coarse resolution. Through the introduction of random finite sets the problem of target initialization and release can be modeled implicitly, thus allowing the filter to track a time-varying number of targets. Target initialization and release is also supported by BraMBLe [7] through a joint filter, which is enhanced with an additional, discrete dimension reporting the number of tracked targets. A very efficient implementation of a joint occlusion-robust likelihood based on Bayesian correlation [8] allows near-real time performance with 3 targets. Mixture tracking is proposed in [9]. Each target is tracked as a single mode of a unique, multimodal, distribution defined on a 1-body state space. In this case, principled occlusion reasoning is not possible since a likelihood cannot be defined as a function of several hypotheses of the same state space. In [10], a MRF motion prior is used to describe interactions between different targets in a joint Bayesian formulation. The importance sampler in the traditional particle filter implementation is replaced with a Markov Chain Monte Carlo (MCMC) sampler which proves to be more efficient. The method lacks an occlusion-robust observation likelihood which is crucial when a top-down view is not available.

A variational approach has been adopted in [11] to avoid track coalescence of identical appearing targets over a similar MRF model, which is employed to implement a spatial exclusion principle. However, these strategies do not consistently handle occlusion, they rather tend to exclude hypotheses representing an occlusion by penalizing them. During long term occlusions such methods are prone to fail. More efficient sampling strategies for high-dimensional problems have been proposed. According to the paradigm of simulated annealing, a set of increasingly peaked likelihood functions is used to explore the state space gradually, thus more efficiently, obtaining promising results when tracking articulated objects on motion capture data [12]. Hyperdynamic sampling [13] is another attempt, that uses local gradient and curvature information to define an importance sampler that tends to explore also the neighborhood of attraction basins. Posterior gradients are also used to improve the high-dimensional MCMC sampler in [14]. Hybrid methods that combine sampling with local optimization are proposed in [15], [16].

Compared with these previous approaches, the proposed algorithm has the advantage of (i) relying on a unified probabilistic framework whose soundness is theoretically assessed and (ii) reasoning explicitly about occlusions according to image formation principles, while (iii) maintaining an affordable computational cost. In addition, it is not tailored to a specific type of measurements and can operate reliably with a single camera.

II. HYBRID JOINT-SEPARABLE FILTER FOR SEQUENTIAL STATE ESTIMATION PROBLEMS

In this paper, object tracking is interpreted as a sequential state estimation problem. Basically, the dynamic components of interest of an environment are described with a vector x of numbers, the *state*, which is evolving in time and observed at discrete times t through measurement vector z_t. The aim is then to estimate its posterior distribution at t (or *belief*) $p(x_t|z_{1:t})$ conditioned on a sequence of observations $z_{1:t}$ gained up to t. In order to support sequential estimation imposed by real time applications, signal x_t is modeled as a first order Markov process, and observations $z_{1:t}$ are assumed to be conditionally independent given a sequence of states $x_{1:t}$. This enables us to compute a new estimate $p(x_t|z_{1:t})$ solely from the actual observation z_t and its previous estimate $p(x_{t-1}|z_{1:t-1})$ in a predict-and-update fashion using stochastic propagation and Bayes law

$$p(x_t|z_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|z_{1:t-1})\ dx_{t-1} \tag{1}$$

$$p(x_t|z_{1:t}) \propto p(z_t|x_t)p(x_t|z_{1:t-1}). \tag{2}$$

Eq. 1-2 define the sequential Bayes filter. Apart from its simplicity, the key feature making this model attractive is that it allows for principled modeling of knowledge about system evolution $p(x_t|x_{t-1})$ and measurement process $p(z_t|x_t)$. An algorithm based on the above recursion can therefore be explicitly tuned to specific behavioral patterns and implement physically based observation models relying on measurement formation principles. The model is fully specified by an initial distribution $p(x_0)$, the dynamical model $p(x_t|x_{t-1})$, and the observation model $p(z_t|x_t)$. A non-parametric implementation, the *particle filter*, is discussed in Sec. V-A.

It is worth to point out that the context addressed by this article imposes the choice of a probabilistic framework. Measurements may convey intrinsic uncertainty which cannot be eliminated due to occlusion, target similarity and background clutter. For efficiency reasons, illumination effects (e.g. shadows) are neglected and only coarse geometric and appearance models can be used, leading to a noisy measurement process. Estimates are therefore inherently inaccurate and a deterministic framework would not adequately account for this.

A. *Multitarget estimation: joint versus separable approach*

The global configuration of a set of targets can be described by a single vector $\mathbf{x}$, the joint state, where its components x^k (which may themselves be vectors) represent the states of the different targets k. Multitarget tracking can be posed as the problem of estimating the distribution of joint configurations with a single filter, the joint filter. By doing so, any kind of interaction both at the propagation level (behavior) and the observation level (occlusions) can be incorporated into the corresponding models $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $p(z_t|\mathbf{x}_t)$. Although powerful due to its generality, this approach suffers from the *curse of dimensionality* (state space dimension proportional to the number of targets) resulting in expensive computations: the representation size as well as the cost of computing the update recursion, grows exponentially with the number of targets.

A simpler solution would be to represent and estimate the evolution of the targets independently, by instantiating a single tracker for each object. The dynamical model and observation model are in this case assumed to be separable, leading to separable posteriors under the assumption that the initial distribution is also separable. While this approach scales linearly with the number of targets, it is blind to any kind of interaction. In particular, occlusions, a major source of failure for tracking systems, are ignored. The *Hybrid Joint-Separable* approach, first introduced in [1] to manage the trade-off between computational complexity and reliable

modeling, is reviewed and extended next.

B. Hybrid joint-separable multitarget filter

The proposed approach is based on an efficient representation of the joint belief by means of its marginal components. More precisely, we assume that the joint distributions involved (priors and posteriors for each time step) are suitably approximated via the outer product of their marginal components

$$p(\mathbf{x}_t|z_{1:\tau}) \approx \prod_k p(x_t^k|z_{1:\tau}) \tag{3}$$

where we define

$$p(x_t^k|z_{1:\tau}) \stackrel{\text{def}}{=} \int p(\mathbf{x}_t|z_{1:\tau})\, d\mathbf{x}_t^{\neg k} \tag{4}$$

with $\mathbf{x}_t^{\neg k}$ representing the vector $\mathbf{x}_t$ with the k-th component removed and τ takes values $t-1$ and t for prior and posterior distributions, respectively. With this notation, the marginal components of predicted distributions (which are obtained by applying Eq. 1) can be rewritten in the following form reminiscent of a separable formulation:

$$\begin{aligned} p(x_t^k|z_{1:t-1}) &= \iint p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|z_{1:t-1})\ d\mathbf{x}_{t-1} d\mathbf{x}_t^{\neg k} \\ &\approx \iint p(\mathbf{x}_t|\mathbf{x}_{t-1}) \prod_h p(x_{t-1}^h|z_{1:t-1})\ d\mathbf{x}_{t-1} d\mathbf{x}_t^{\neg k} \\ &= \iint p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}^{\neg k}|z_{1:t-1}) d\mathbf{x}_{t-1:t}^{\neg k}\, p(x_{t-1}^k|z_{1:t-1}) dx_t^k \end{aligned} \tag{5}$$

where $\mathbf{x}_{t-1:t}^{\neg k}$ denotes the vector composed of the previous and the current substate, $\mathbf{x}_{t-1}^{\neg k}$ and $\mathbf{x}_t^{\neg k}$, respectively. Similarly, Bayes filter Eq. 2 provides marginal posteriors as follows:

$$\begin{aligned} p(x_t^k|z_{1:t}) &\propto \int p(z_t|\mathbf{x}_t) p(\mathbf{x}_t|z_{1:t-1}) d\mathbf{x}_t^{\neg k} \\ &\approx \int p(z_t|\mathbf{x}_t) \prod_h p(x_t^h|z_{1:t-1}) d\mathbf{x}_t^{\neg k} \\ &= \int p(z_t|\mathbf{x}_t) p(\mathbf{x}_t^{\neg k}|z_{1:t-1}) d\mathbf{x}_t^{\neg k}\ p(x_t^k|z_{1:t-1}). \end{aligned} \tag{6}$$

The *Hybrid Joint-Separable multitarget filter* is then described by a separable initial distribution

$$p(\mathbf{x}_0) = \prod_k p(x_0^k) \tag{7}$$

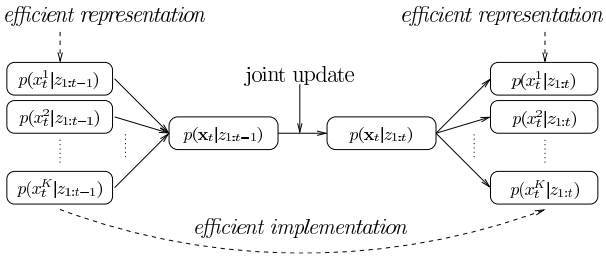

Fig. 1. HJS recursion with joint models: joint prior distribution is first reconstructed as the product of marginal beliefs, then updated using joint dynamical and observation models and finally marginalized for each object, obtaining marginal posterior distributions. An efficient implementation would compute marginal posteriors directly from marginal priors without any explicit joint reconstruction, possibly preventing exponential computational complexity in the number of targets.

and a two-step recursion

$$p(x_t^k|z_{1:t-1}) = \int p(x_t^k|x_{t-1}^k)p(x_{t-1}^k|z_{1:t-1})\ dx_{t-1}^k \tag{8}$$

$$p(x_t^k|z_{1:t}) \propto p(z_t|x_t^k)p(x_t^k|z_{1:t-1}) \tag{9}$$

whose *marginal dynamical* and *observation models* are defined according to

$$p(x_t^k|x_{t-1}^k) = \iint p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}^{\neg k}|z_{1:t-1})d\mathbf{x}_{t-1:t}^{\neg k} \tag{10}$$

$$p(z_t|x_t^k) = \int p(z_t|\mathbf{x}_t)p(\mathbf{x}_t^{\neg k}|z_{1:t-1})d\mathbf{x}_t^{\neg k}. \tag{11}$$

It therefore defines a hybrid between a joint and a separable formulation, consisting in a separable belief representation while relying on updates with joint process models.

An important property of the HJS model is its economical representation size, which grows linearly with the number of objects: representation complexity is the same as for fully separable 1-body models. As far as computational cost is concerned, Fig. 1 shows the different steps involved when computing the recursion defined in Eq. 8-9 in a straightforward way. The key to an efficient implementation of the HJS filter rests on an algorithm that directly updates the marginal distributions while accounting for joint dynamics and observation models, without explicitly carrying on estimation in the joint domain. Whether this is possible or not depends on the structural properties of the joint models $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $p(z_t|\mathbf{x}_t)$, which in some cases may be imposed by approximations (Sec. III) or derive from the process itself (Sec. IV).

C. Hybrid joint-separable model and optimal reconstruction

The estimation framework just presented is based on the approximation stated in Eq. 3. It can be justified by assuming statistical independence of random variables x_t^k after prediction and after observation. This is a much weaker assumption than the one underlying a separable formulation, where the signals $x^k(t)$ themselves, as well as the observation process, are assumed to be independent [1]. The proposed approach then addresses a significantly richer class of models. Its soundness is now assessed in terms of optimal function approximation.

As joint estimation is too expensive to perform, we might limit ourselves to carry on estimation in single target spaces, under the condition that guarantees us to do this *as best we can*. Formally, this means that we are looking for a mapping, or projection, from the joint probability space to the product space of single target distributions which optimizes some function that measures reconstruction quality in joint space:

Problem 1: Given the outer product reconstruction $\mathcal{R}$

$$\mathcal{R} : \mathcal{P}(\mathcal{X}^1) \times \cdots \times \mathcal{P}(\mathcal{X}^K) \;\longmapsto\; \mathcal{P}(\mathbf{X})$$
$$\mathcal{R}(p(x^1), \cdots, p(x^K)) = \prod_k p(x^k)$$

find the mapping $\mathcal{M}$

$$\mathcal{M} : \mathcal{P}(\mathbf{X}) \;\longmapsto\; \mathcal{P}(\mathcal{X}^{\mathbf{1}}) \times \cdots \times \mathcal{P}(\mathcal{X}^{\mathbf{K}})$$

that minimizes a suitable distance between a joint distribution $p(\mathbf{x})$ and its reconstructed projection $\mathcal{R}(\mathcal{M}(p(\mathbf{x})))$.

From a functional point of view, the HJS filter can be regarded as a joint filter where each prediction or measurement update is followed by a *compression* carried out by applying $\mathcal{R} \circ \mathcal{M}$. As a natural measure of distance between distributions the Kullbach-Leibler (KL) divergence [17] can be used

$$\mathcal{D}(p_1 \| p_2) = \int p_1(\mathbf{x}) \ln \frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \, d\mathbf{x}. \tag{12}$$

Intuitively, it tells us how much we loose in terms of average information if we assume that $\mathbf{x}$ is distributed according to p_2 when it belongs to p_1. The following theorem states that the HJS model is optimal in this sense, thus solving Problem 1.

[1] In this case interactions need to be resolved by a higher level process, e.g. a Multiple Hypothesis tracker (MHT) or a data association filter (JPDAF).

Theorem 1: HJS model is optimal w.r.t. joint reconstruction quality in the KL divergence sense.

Proof: KL divergence between a joint distribution $p(\mathbf{x})$ and its $\mathcal{R}\circ\mathcal{M}$-compressed version is given by

$$\begin{aligned}\mathcal{L}(p) = \mathcal{D}(p\|(\mathcal{R}\circ\mathcal{M})(p)) &= \int p(\mathbf{x}) \ln \frac{p(\mathbf{x})}{\prod_k p(x^k)}\, d\mathbf{x} \\ &= \int p(\mathbf{x}) \ln p(\mathbf{x})\, d\mathbf{x} - \sum_k \int p(\mathbf{x}) \ln p(x^k)\, d\mathbf{x} \\ &= -\mathcal{E}(p) - \sum_k \int p(\mathbf{x}) \ln p(x^k)\, d\mathbf{x}\end{aligned}$$

where $\mathcal{E}(p)$ denotes the differential entropy of distribution p, which is independent of $\mathcal{M}$. Using the method of Lagrange multipliers to impose the constraint of $p(x^k)$ being distributions (they sum up to 1), we can look for minima of the functional Lagrangian $\mathcal{L}(\mathbf{p}, \boldsymbol{\lambda})$ given by

$$-\sum_k \Big(\int p(\mathbf{x}) \ln p(x^k)\, d\mathbf{x} + \lambda_k (1 - \int p(x^k)\, dx^k) \Big).$$

The saddle points of $\mathcal{L}(\mathbf{p}, \boldsymbol{\lambda})$ are found in

$$\begin{aligned}0 &= \frac{\partial}{\partial p(x^k)} \Big(\int p(\mathbf{x}) \ln p(x^k)\, d\mathbf{x} - \lambda^k \int p(x^k)\, dx^k \Big) \\ &= \frac{\partial}{\partial p(x^k)} \int \ln p(x^k) \int p(\mathbf{x})\, d\mathbf{x}^{\neg k} - \lambda^k p(x^k)\, dx^k \\ &= \int \Big(\frac{1}{p(x^k)} \int p(\mathbf{x})\, d\mathbf{x}^{\neg k} - \lambda^k \Big)\, dx^k\end{aligned}$$

a property that must hold regardless the instance of $p(\mathbf{x})$. Thus, marginal projection follows. ■

D. Relation to Belief Propagation and Mean Field methods

Recently, there has been growing interest in developing approximate inference algorithms on graphical models as arising in Vision applications [11], [18]–[21]. When the problem at hand allows to state conditional independence assumptions between nodes so that a Bayesian net becomes structured, Belief Propagation (BP) provides an attractive inference engine. It operates by passing local messages between pairwise coupled nodes which encode statistical dependence among neighbors and their observations, so that e.g. marginals over the nodes can be estimated with linear complexity. In acyclic nets this solution is found to be exact (Bayes filter performs BP over a Markov chain), while when loops exist iterative message passing leads to estimates that are

still optimal, at least locally, as far as convergence is achieved [22]. While originally developed for Gaussian models over discrete states, BP has recently been formulated in more general, non-parametric settings [23], [24]. Efficiency of BP comes from a factored representation that allows to organize expensive joint computations in terms of smaller local computations. While such a decomposition may be plausible when modeling interactions among targets (Sec. III), there is no consistent way to pairwise decompose a physically based model of the visual observation process; this would be equivalent to fuse likelihoods obtained by considering only one occlusion at a time [21] which is not consistent with multiple occlusions. This rules out this class of methods when addressing principled occlusion handling since there is no gain in running BP on a jointly connected graph. One may partially overcome this limitation by adapting the graph structure dynamically [25], e.g. to account for possible interactions only where they are suggested by prediction [11], [26]. The so obtained dynamical Bayesian net conveys additional structure, but cliques of occluding targets must still be represented jointly, where message generation remains costly.

To break down the computational load in graphical models with large cliques, variational methods can be used. The basic idea is to carry out inference in a tractable subspace of the joint probability space [27]. In the mean field approach the subspace chosen is that of factored distributions, much like by the HJS model (Eq. 3). The variational approximation q is found by solving a set of fixed-point equations [11], [28] iteratively, which are designed to optimize its KL-divergence fit to the original (joint) distribution. This is indeed a solution to Problem 1, which, however, is different to the one we propose: standard variational methods optimize KL-divergence of p with respect to the variational distribution q, i.e. $\mathcal{D}(q\|p)$, while in Theorem 1 we state the solution for optimality of $\mathcal{D}(p\|q)$. In our opinion, both formulations have the same reason to exist; in addition, our solution is build upon true marginals. While we have obtained a closed form solution whose evaluation complexity is still exponential for generic models, optimizing $\mathcal{D}(q\|p)$ does provide no closed form solution but a set of equations suitable for implementation of a fixed-point algorithm to optimize single components one at a time, thus efficiently. The principal claim of this paper is that computing exact posterior marginals (i.e. the adopted variational distribution) obtained by observing a separable prior through an occlusion process can be done efficiently, in quadratic complexity. But while BP and the mean field method are powerful tools for approximate inference with generic models, this solution is specific to the

occlusion problem.

In conclusion, the approach taken in this paper can be classified as a mean field method over a dynamical Bayesian net featuring explicit occlusion reasoning and mutual spatial exclusion. An efficient closed form solution is provided for this specific problem (Sec. IV), coming from a mathematical derivation that exploits the probabilistic structure imposed by the occlusion process. In this sense, the HJS filter performs exact inference on an approximate model for the visual multibody tracking problem.

III. HJS MOTION PRIOR

Even though visual tracking mostly relies on reliable modeling of the measurement process, in some applications it might be appropriate to account for interactions between targets at the level of dynamical model. This has been addressed e.g. in [10], [11] to avoid track coalescence of targets with similar appearance. This section builds upon the ideas proposed there, and shows how they can be imported into the HJS framework.

A. Forward dynamical model

The marginal dynamical model in Eq. 10 is universal, in the sense that model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ may capture any kind of interaction. For example, it may describe how velocity components of the same object relate to each other during evolution, which may be regarded as a strong interaction since they are correlated by physical constraints. On the other hand, a small group of people that is visiting a museum together tends to remain compact, resulting in weak interaction among their positions. For the latter class of interactions, where for reasonable high observation rates the dynamic component of the propagation process underlying $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is dominated by independent 1-target laws q, a joint factor may be added afterwards. The *forward model* is then defined according to

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) \approx p(\mathbf{x}_t) \prod_k q(x_t^k|x_{t-1}^k) \tag{13}$$

which enables us to express the marginal prior in Eq. 8 as

$$p(x_t^k|z_{1:t-1}) = \int p(\mathbf{x}_t) p^-(\mathbf{x}_t|z_{1:t-1}) \; d\mathbf{x}_t^{\neg k} \tag{14}$$

with p^- denoting the independently propagated joint prior

$$p^-(\mathbf{x}_t|z_{1:t-1}) = \prod_h \int q(x_t^h|x_{t-1}^h) p(x_{t-1}^h|z_{1:t-1}) dx_{t-1}^h. \tag{15}$$

Note that restricting ourselves to dynamical models of this form does not prevent us from considering interactions with a dynamic component: $\mathbf{x}_t$ may itself contain dynamic components (e.g. target velocities as in Sec. VI) which can be taken into account by $p(\mathbf{x}_t)$.

B. Belief Propagation with MRF interaction model

The advantage of the forward formulation in Eq. 13 lies in the fact that (i) interaction is no longer temporal, thus less complex and easier to model, and (ii) the complexity of the propagation step is reduced (the double joint-space integral in 10 becomes, by means of Eq. 8, a single one in Eq. 14 plus K single space integrals in Eq. 15). Nonetheless, computing Eq. 14 with a fully joint model $p(\mathbf{x}_t)$ is still expensive. In many cases, however, it is sufficient to define $p(\mathbf{x}_t)$ in terms of a MRF. In a MRF a joint model is factored into the product of local potential functions which account for dependencies among subsets of random variables, or *neighborhood cliques* [29]. The complexity associated with Eq. 14 still depends on which pattern has to be modeled. With a pairwise MRF as in [10], [11], the graphical representation on which we want to compute marginals becomes

$$p(\mathbf{x}_t|z_{1:t-1}) = \prod_{\langle h,k\rangle\in\mathcal{E}} p(x_t^h, x_t^k) \prod_h p^-(x_t^h|z_{1:t-1}) \tag{16}$$

with $\mathcal{E}$ denoting the set of interacting node pairs. $\mathcal{E}$ can be adapted dynamically to reduce complexity, as discussed in Sec. II-D. We apply Belief Propagation (BP) [22] to compute approximate marginals efficiently. At each iteration i, a new set of functions $m_i^{h,k}(x_t^k)$, the *messages*, are computed according to

$$m_i^{h,k}(x_t^k) = \int p(x_t^h, x_t^k) p^-(x_t^h|z_{1:t-1}) \prod_{n\in\mathcal{C}_h\setminus k} m_{i-1}^{n,h}(x_t^h)\, dx_t^h \tag{17}$$

with $\mathcal{C}_k = \{h \mid \langle h,k\rangle \in \mathcal{E}\}$ denoting the *neighborhood* of node k. An approximation to the marginal beliefs is then found by

$$p(x_t^k|z_{1:t-1}) \;\approx_i\; p^-(x_t^k|z_{1:t-1}) \prod_{h\in\mathcal{C}_k} m_i^{h,k}(x_t^k). \tag{18}$$

We accept approximation i if its KL-divergence to the approximation obtained at $i-1$ is below a predefined threshold. In Sec. V we discuss an efficient implementation of BP when $p^-(x_t^k|z_{1:t-1})$ and $p(x_t^h, x_t^k)$ are non-parametric. In Sec. VI-C we show that with $p(x_t^h, x_t^k)$ implementing spatial exclusion (i.e. imposing x_t^h and x_t^k not to occupy the same space at the same time) permits reliable

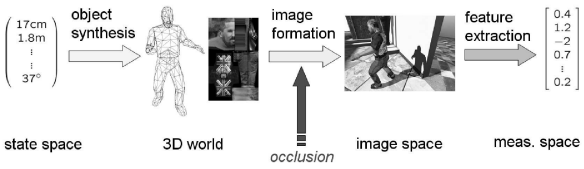

Fig. 2. Measurement formation process: a numeric description of the environment is first mapped onto 3D geometry and appearance of constituent objects, then a picture is generated according to image formation principles and finally a measurement vector is extracted. Occlusions are generated during world-to-image projection.

tracking through long term occlusions even if targets appear identical. More sophisticated models can also be designed which may e.g. consider expected impact time for collision avoidance or target congregation [30].

IV. HJS LIKELIHOOD FOR VISUAL TRACKING

Reliable occlusion handling among targets has been shown to be a hard problem for real time tracking systems. While approaches exist that deal with this problem in a principled fashion [2], [5], [7], [10], they all suffer from exponential complexity blowup as the number of targets increases. As shown in [1], the HJS model allows an exact, while still scalable, solution to this problem that allows real time tracking of several bodies. However, the underlying 1D formulation conveys an occlusion relation that is discrete, leading to a model that can explain only complete occlusions but not partial ones. In this section we show how this limitation can be overcome, by carrying on occlusion reasoning at pixel level on top of a fully image based likelihood, while still keeping the same, quadratic, order of complexity.

A. Physically based multitarget observation model

The role of the observation model $p(z|\mathbf{x})$ is to quantify the likelihood of a measure z given that the observed world is in state $\mathbf{x}$. These two vectors must then be mapped onto a common space, the feature space. A natural way to define this mapping is to mimic the real measurement formation process as described in Fig. 2. If 3D geometric and appearance models of scene objects are available (or have been acquired during model acquisition stage), hypothesis $\mathbf{x}$ can be rendered into a synthetic picture $g(\mathbf{x})$ by Computer Graphics principles. Mapping g is called *rendering function* and implements a model of the imaging process. Image $g(\mathbf{x})$ may then be

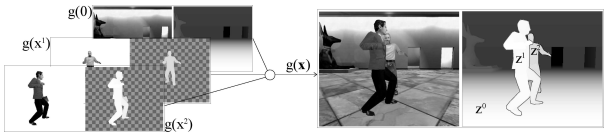

Fig. 3. Joint rendering procedure. Single object rendering outputs are combined at each pixel, considering camera closeness. Background is modeled through an additional, constant layer.

compared with real image z by means of a heuristic distance D between suitably extracted feature vectors

$$p(z|\mathbf{x}) \propto \exp(-D^2(z, g(\mathbf{x}))/2\sigma^2) \tag{19}$$

thus quantifying their similarity, or likelihood. Parameter σ can be used to control the width of this distribution; a principled way to assign σ when the background is known is proposed in [15]. The choice of representation (feature space) and metric D is crucial to $p(z|\mathbf{x})$, and application dependent. Modeling unconstraint human appearance is complex, and a research challenge by its own [31], [32]. We will specialize to color histograms (Sec. IV-B.1) and Bhattacharyya-coefficient based metric (Sec. VI-A). With this choice $p(z|\mathbf{x})$ is evaluated on spatial statistics, thus making it a smooth function of $\mathbf{x}$ [33]; in this way we prevent from well known problems related to sharply peaked and irregular likelihoods [15]. Nonetheless, the derivation of the method is kept completely general and thus other appearance descriptors may be used as well.

A closer look to image formation reveals that occlusions are generated during perspective projection from 3D world to 2D image space according to camera closeness. It follows that, by neglecting secondary interactions such as shadows or specular effects, rendering function $g(\mathbf{x})$ may be decomposed into single object components $g(x^k)$ whose outputs are combined at each pixel according to camera distance (Fig. 3). Hence, at pixel u we have

$$g_u(\mathbf{x}) = \begin{cases} g_u(x^k) & \text{if } x^k \underset{u}{<} x^h \quad \forall h \neq k \\ g_u(\mathbf{x}^{\neg k}) & \text{elsewhere} \end{cases} \tag{20}$$

where $x^k <_u x^h$ states that, at u, x^k is closer to the camera than x^h, a property that can be derived from 3D object geometry, and $g(\emptyset)$ is the background. Thus, once the image regions z^k

belonging to the different objects have been identified using the shape model [2], their appearance no longer depend on each other. In other words, the observation model is separable in image space (where D^2 is additive)

$$p(z|\mathbf{x}) = p(z^0|\mathbf{x}) \prod_k p(z^k|\mathbf{x}). \tag{21}$$

This defines an observation model that is robust to partial and complete occlusions and complies with statistical independence of object blobs. Similar models have been proposed for visual localization and tracking [7], [8], [12], [20].

B. HJS likelihood

The separable formulation in Eq. 21 leads to an approximation of Eq. 11 that can be implemented efficiently, in a way reminiscent of the derivation for the targets on a single line-of-sight in [1]. Manipulation of the integral in Eq. 11 is possible in log-likelihood domain: the problem translates to finding an efficient algorithm for computing

$$l(z_t|x_t^k) = \int l(z_t|\mathbf{x}_t) p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \tag{22}$$

where the joint log-likelihood function $l(z|\mathbf{x})$ is given by

$$l(z|\mathbf{x}) \equiv D^2(z, g(\mathbf{x})) \propto -\log p(z|\mathbf{x}).$$

Its closeness to the original formulation is assessed next.

Theorem 2: Expressing the marginal observation model of Eq. 21 in log-likelihood domain as by Eq. 22 is equivalent to a first order approximation.

Proof: By replacing the joint likelihood $p(z_t|\mathbf{x}_t)$ with the first order term $1 - l(z_t|\mathbf{x}_t)$ of its Taylor expansion evaluated in $l = 0$ one obtains:

$$\begin{aligned} p(z_t|x_t^k) &\propto \int e^{-l(z_t|\mathbf{x}_t)} p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ &\approx \int \big(1 - l(z_t|\mathbf{x}_t)\big)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ &= 1 - \int l(z_t|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k}. \end{aligned}$$

[2]To obtain z^k on real images we will assume calibrated camera and generalized-cylinder model [7]. Both requirements can be relaxed: calibration is not needed if x contains explicit depth information such as target scale (this can be used to extract perspective camera distance); 3D shape can be replaced by a 2D model which may also be adapted online.

This latter term can be interpreted as the first order term of the Taylor expansion of the exponential of

$$-\int l(z_t|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k}$$

which proves the theorem. ■

Separability in image space (Eq. 21) allows to decompose the marginal log-likelihood into the sum of two terms

$$\begin{aligned} l(z_t|x_t^k) &= \int l(z_t|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ &= \int \Big(l(z_t^0|\mathbf{x}_t) + \sum_h l(z_t^h|\mathbf{x}_t) \Big)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ &= \int l(z_t^k|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ &\qquad + \int l(z_t^{\neg k}|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \end{aligned} \tag{23}$$

whose interpretation is as follows:

- the foreground term, or presence log-likelihood

$$\int l(z_t^k|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \tag{24}$$

  accounts for the contribution of the expected to be visible parts of target k;

- the background term, or occlusion evidence

$$\int l(z_t^{\neg k}|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \tag{25}$$

  accounts for the accumulated likelihood of those image regions in which target k, locked to x_t^k, is expected either to be covered by at least one of the other targets or not to be visible at all (pixels belonging to the background).

We show now how these terms may be calculated efficiently.

1) Single target appearance log-likelihood: For the sake of clarity we will now choose a specific class of log-likelihood functions, namely those based on a distance, d, between color distributions. For a given joint state $\mathbf{x}$ the 1-body log-likelihood $l(z^k|\mathbf{x})$ is taken as the distance between the color histogram extracted from the image region z^k (identified by the adopted shape models rendered at $\mathbf{x}$) and its pre-acquired color model. Alternative representations can be chosen [2], [8], [31] which may lead to more efficient, but less descriptive likelihoods [7]; HJS

tracker provides an efficient engine that enables to use rich descriptors, such as color histograms, which allow to distinguish between different targets of the same shape. 1-body log-likelihood is then defined as

$$l(z^k|\mathbf{x}) = \frac{\alpha^k d(\mathrm{Pr}_z^k, \mathrm{Pr}_g^k) + \beta^k d_{occ}}{\alpha^k + \beta^k} \tag{26}$$

with α^k, β^k the number of visible and hidden silhouette pixels

$$\alpha^k = \int \delta(x^k \underset{u}{\lessdot} \mathbf{x}^{\urcorner k})\ du, \quad \alpha^k + \beta^k = \int \delta(x^k \underset{u}{\lessdot} \infty)\ du$$

and d_{occ} a parameter to be assigned. Binary mask $\delta(x^k <_u \mathbf{x}^{\urcorner k})$ takes value 1 where x^k is the closest to the camera and 0 elsewhere. Eq. 26 linearly interpolates the evidence obtained by comparing normalized measured ($\zeta = z$) and rendered ($\zeta = g(x^k)$) histograms

$$\mathrm{Pr}_\zeta^k(c) = \frac{1}{\alpha^k} \int \delta(x^k \underset{u}{\lessdot} \mathbf{x}^{\urcorner k}) \delta_c(\zeta_u)\ du \tag{27}$$

with a penalty term d_{occ} assigned to unobserved body parts. $\delta_c(\zeta_u)$ takes value 1 where $\zeta_u \equiv c$ and 0 elsewhere. It therefore allocates a likelihood to the projection of the entire body, not only to the image portion in which it is visible. This is different from conventional formulations [8] and is justified as follows. Let us consider an image of two non occluding targets, and let $\mathbf{x}_1$ be the correct configuration and $\mathbf{x}_2$ be a hypothesis in which target A is correctly localized while target B is hypothesized in the occlusion volume of A. The contribution of B to the likelihood of $\mathbf{x}_2$ would be null when only the visible part of the silhouette is taken into account. Now, if background information is discarded, only B is contributing to the likelihood of $\mathbf{x}_2$. This means that if the noise in the appearance model of A is greater than the one of B, $\mathbf{x}_2$ would be assigned a higher likelihood than the correct configuration $\mathbf{x}_1$. A tracker that discards background information (i.e. information about where the targets cannot be) is therefore prone to fail when B crosses the occlusion volume of A since most of the probability mass would be absorbed by occluded hypotheses and never released to the correct ones, producing a *phantom mode* in the belief. This artifact is associable to the missed demand for true Bayesian inference [8] which requires the observations to be constant and not a function of $\mathbf{x}$ as when omitting the background (later on we will focus on such background independent models). With the likelihood in Eq. 26 this problem is alleviated since $d_{occ} > 0$ penalizes occluded hypotheses. Parameter d_{occ} should be assigned a value between the responses of d over the background and on a noisy version of the model (as noisy as expected on the sequence to be processed).

2) Computation of the foreground term: Let $\bar{w}^k_{\mathbf{x}}(u)$ be the function of (continuous) pixel coordinates u defined as follows

$$\bar{w}^k_{\mathbf{x}}(u) = \delta(x^k \underset{u}{\leq} \mathbf{x}^{\urcorner k}). \tag{28}$$

The color distribution extracted from image ζ now writes as

$$\mathrm{Pr}^k_\zeta(c) \propto \int w(u)\delta_c(\zeta_u)\ du, \tag{29}$$

where the generic, continuous-valued map w is evaluated at $w = \bar{w}^k_{\mathbf{x}}$. Map w thus assigns a weight to each pixel to be accounted for when computing the histogram. We now assume that d is homogeneous, i.e. that α^k cancels out in Eq. 26, 27. According to Eq. 26, foreground term $l(z^k|\mathbf{x})$ can now be understood as a function of this map, i.e. a functional of w

$$\mathcal{F}(w) = d\Big(\int w(u)\delta_c(z_u)\ du, \int w(u)\delta_c(g_u(x^k))\ du\Big) \tag{30}$$

evaluated at $w = \bar{w}^k_{\mathbf{x}}$. From the theory of variational calculus we know that it is possible to generate a polynomial expansion of $\mathcal{F}(w)$ in a form reminiscent of the Taylor's series known from function analysis. Taylor expansion of functional $\mathcal{F}(w)$ evaluated at a selected function w_0 is given by

$$\mathcal{F}(w) = \mathcal{F}(w_0) + \int \frac{\partial \mathcal{F}}{\partial w}(w_0)(w - w_0)\ du + \mathcal{O}^2(w - w_0).$$

By replacing $l(z^k_t|\mathbf{x}_t)$ in Eq. 24 with its first order expansion we obtain the following approximation of the foreground term:

$$\begin{aligned}\int l(z^k_t|\mathbf{x}_t)\ p(\mathbf{x}_t^{\urcorner k}|z_{1:t-1})\, d\mathbf{x}_t^{\urcorner k} &= \\ \approx \int\Big[\mathcal{F}(w_0) + \int\frac{\partial\mathcal{F}}{\partial w}(w_0)(\bar{w}^k_{\mathbf{x}_t} - w_0)\ du\Big]p(\mathbf{x}_t^{\urcorner k}|z_{1:t-1})\ d\mathbf{x}_t^{\urcorner k} &\\ = \mathcal{F}(w_0) + \int \frac{\partial\mathcal{F}}{\partial w}(w_0)(\int \bar{w}^k_{\mathbf{x}_t}\ p(\mathbf{x}_t^{\urcorner k}|z_{1:t-1})\, d\mathbf{x}_t^{\urcorner k} - w_0)\ du.&\end{aligned} \tag{31}$$

The right-hand side of the above equation is exactly the first order expansion of functional $\mathcal{F}(w)$ evaluated at

$$w^*_{x^k_t} = \int \bar{w}^k_{\mathbf{x}_t}\ p(\mathbf{x}_t^{\urcorner k}|z_{1:t-1})\, d\mathbf{x}_t^{\urcorner k}. \tag{32}$$

Hence, to first order of accuracy, the foreground term can be evaluated at once as the distance between properly weighted color distributions

$$d\Big(\int w^*_{x^k_t}(u)\delta_c(z_u)\ du, \int w^*_{x^k}(u)\delta_c(g_u(x^k))\ du\Big). \tag{33}$$

The advantage of rewriting the foreground term in this form lies in the transfer of the marginalization operation from log-likelihoods $l(z_t^k|\mathbf{x}_t)$ to their masks $\bar{w}^k_{\mathbf{x}_t}$. The so obtained map $w^*_{x_t^k}$ acts then as a *soft image partition* for histogram extraction. In contrast to the original formulation of Eq. 24 which would require the evaluation of $l(z_t^k|\mathbf{x}_t)$ for all possible combinations $\mathbf{x}_t^{\urcorner k}$, marginalization of masks $\bar{w}^k_{\mathbf{x}}$ can be done efficiently. To see this let us recall that the temporal prior, $p(\mathbf{x}_t^{\urcorner k}|z_{1:t-1})$, is separable. Simple manipulations then lead to

$$w^*_{x_t^k}(u) = \begin{cases} \prod_{h \neq k} \left(1 - \int_{x_t^h \underset{u}{\leq} x_t^k} p(x_t^h|z_{1:t-1})\, dx_t^h\right) & \text{if } x_t^k \underset{u}{<} \infty \\ 0 & \text{elsewhere} \end{cases} \tag{34}$$

Hence, occlusion map w^* for target k factors pixel-wise into one component per other target h. It is easy to show that each factor hereby expresses the prior probability that target h is not visible in the considered pixel u. Provided that the relative order of states does not change with pixel coordinates (a property that holds for convex, impenetrable bodies), each state x_t^h contributes to w^* with its silhouette, $\Delta_g(x_t^h)$, accumulated onto w^* with intensity equal to its prior:

$$\begin{aligned} \int_{x_t^h \underset{u}{\leq} x_t^k} p(x_t^h|z_{1:t-1})\, dx_t^h &= \int \delta(x_t^h \underset{u}{\leq} x_t^k) p(x_t^h|z_{1:t-1})\, dx_t^h \\ &= \int_{x_t^h < x_t^k} \left[\Delta_g(x_t^h)\right]_u p(x_t^h|z_{1:t-1})\, dx_t^h. \end{aligned} \tag{35}$$

Here $[\Delta]_u$ stays for the value of Δ in pixel u. It has been possible to express Eq. 32-35 in such a way that all integration domains are confined to states that are not farther from the camera than candidate x_t^k. This observation is the key to an efficient implementation: while visiting the state space from camera near to camera far states, occlusion map w^* can be computed incrementally, while accumulating silhouette images. The implementation inside a particle filter is discussed in Sec. V-B.2. A typical occlusion map is shown in Fig. 4.

3) Computation of background term: The background term in Eq. 25 accounts for contributions due to image region $z_t^{\urcorner k}$ which is complementary to z_t^k, whose support is therefore composed of visible silhouette parts of other objects plus background. Moreover, silhouettes of the other objects may be only partially covered by the silhouette of state x_t^k under analysis, or even not occluded at all by it. This prevents from evaluating Eq. 25 while visiting the state space from camera near to camera far states because any hypothesis of another object located behind x_t^k (whose likelihood would therefore be calculated afterwards) may contribute. This problem

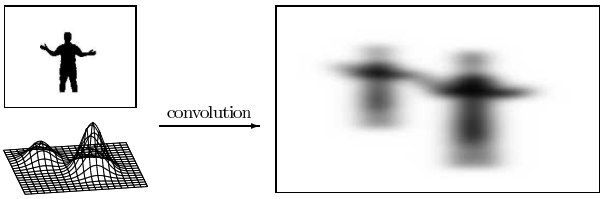

Fig. 4. On the left a template image of the silhouette of a specific pose and a bimodal distribution representing a possible estimate of the person's position is shown. The right image shows the corresponding occlusion map at ∞: the template silhouette becomes blurred due to estimation uncertainty.

can be circumvented when accounting only for the part of the image delimited by the silhouette belonging to x_t^k. In other words, the likelihood of x_t^k is considered to be influenced only by the pixels in which it might be potentially observable. No background measurement is included, and the observed 3D space is reduced to the visual cone that is generated by the candidate silhouette. With this choice, Eq. 23 considers only states x_t^h which are closer to the camera than x_t^k, enabling again incremental evaluation. To see this, let us rewrite the latter term of the marginal log-likelihood in Eq. 23 as a sum of one term per other target h

$$\begin{aligned}\int l(z_t^{\neg k}|\mathbf{x}_t)\; p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} = \\ = \sum_{h\neq k} \int l(z_t^h|\mathbf{x}_t) p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} \\ = \sum_{h\neq k} \int p(x_t^h|z_{1:t-1}) \int l(z_t^h|\mathbf{x}_t^{\neg k}) p(\mathbf{x}_t^{\neg kh}|z_{1:t-1})\; d\mathbf{x}_t^{\neg kh} dx_t^h.\end{aligned} \tag{36}$$

The inner integral on the right-hand side has the same form as the foreground term in Eq. 24, with index h replacing k, and with target k removed (as a consequence of reduced observation space, we have $l(z_t^h|\mathbf{x}_t) = l(z_t^h|\mathbf{x}_t^{\neg k})$). Thus, the inner integral of each term can be calculated as with Eq. 33, by neglecting the presence of object k, already while visiting x_t^h which is, indeed, closer to the camera than x_t^k. The contribution of object h to Eq. 36 can then be calculated incrementally, by accumulating these foreground terms accordingly weighted with their priors. Note that if $w^* \equiv 1$ on the silhouette of x_t^k there is no occlusion hypothesis supported by the priors (recall Eq. 34). It follows that Eq. 36 gives 0 ($p(x_t^h|z_{1:t-1}) \equiv 0$ where $x_t^h <_u x_t^k$), thus HJS likelihood of x_t^k is merely determined by the foreground term. Evaluation of background term can then be bypassed, and the algorithm complexity becomes linear in the number of targets (as

for independent trackers). In other words, dependencies can be detected online, by analyzing w^* when assembled for foreground term computation. The HJS filter then infers over a dynamical Bayesian net (DBN). This resembles also the subordination strategy employed in [3].

In the above derivation we have omitted the fact that the inner integrals in Eq. 36 are calculated on the visual cone of hypothesis x_t^h which might differ from the one of x_t^k. They may overlap completely, partially or may not intersect at all. We need to account for this because otherwise an unlikely hypothesis of another object whose silhouette covers only a single pixel of the candidate would heavily affect its marginal likelihood. If the log-likelihood is interpreted as a measure of mismatch it makes sense to distribute this "error" uniformly on the image region identified by the silhouette of x_t^h. When calculating its contribution to the marginal value of x_t^k only the errors spread on the observed pixels are to be taken into account. In other words, each state x_t^h does no longer contribute with its log-likelihood value $l(z^h|\mathbf{x})$ previously evaluated on the entire silhouette, but only in proportion to its overlap with the silhouette of the candidate x_t^k. Hence, these values must be scaled according to the relative area of intersection

$$r(x^h, x^k) = \int \delta(x^h \underset{u}{\leq} x^k \underset{u}{\leq} \infty)\ du \ \Big/ \int \delta(x^h \underset{u}{\leq} \infty)\ du \tag{37}$$

and the contribution of object h to Eq. 36 becomes

$$\int p(x_t^h|z_{1:t-1}) r(x_t^h, x_t^k) \int l(z_t^h|\mathbf{x}_t^{\neg k}) p(\mathbf{x}_t^{\neg kh}|z_{1:t-1})\ d\mathbf{x}_t^{\neg kh}\ dx_t^h. \tag{38}$$

The implementation details of the algorithm are described in Sec. V-B.2. Here we will only anticipate that Eq. 38 can be computed efficiently when evaluated in image space as follows. While visiting the state space in camera distance order, an image buffer, b_{bg}^k, is filled in by accumulating silhouette images belonging to the states of the other objects, in a similar way as done for w^*. This time the intensity of the silhouette associated with each state x_t^h is given by the product of its prior and the normalized value of the inner integral in Eq. 38:

$$p(x_t^h|z_{1:t-1}) \int l(z_t^h|\mathbf{x}_t) p(\mathbf{x}_t^{\neg kh}|z_{1:t-1})\ d\mathbf{x}_t^{\neg kh} \Big/ \int \delta(x^h \underset{u}{\leq} \infty)\ du. \tag{39}$$

Eq. 25 can then be evaluated at x_t^k by summing up the accumulated intensities contained in the region of b_{bg}^k covered by the candidate silhouette, i.e.

$$\int l(z_t^{\neg k}|\mathbf{x}_t)\ p(\mathbf{x}_t^{\neg k}|z_{1:t-1})\, d\mathbf{x}_t^{\neg k} = \int \delta(x^k \underset{u}{\leq} \infty)\, b_{\text{bg}}^k(u)\ du. \tag{40}$$

A final note on a possible improvement: rather than keeping the intensity constant on the entire silhouette, one can modulate it at each pixel by multiplying Eq. 39 with a normalized version of $w^*_{x_t^h}$. This would have the effect of distributing more error on pixels that are less likely occluded. Consequently, a hypothesis of target k that is covered only by a likely occluded body part of target h would be less penalized.

V. HJS PARTICLE FILTERING

When mapped onto a calculator, Bayes filter based algorithms estimate a numeric description of the involved distributions, whose nature may be parametric (Kalman filtering), semi-parametric (grid-based methods) or non-parametric (particle filtering). The latter class of methods has the advantage of being based on universal approximations while still compact in description, and are not subject to any constraints of the models. This makes them particularly suitable for the implementation of the theory presented.

A. Background

The idea underlying particle filters is to maintain a compressed representation of the belief by means of a set of representative sample states, the particles. In perfect MonteCarlo sampling, these samples are chosen independently and identically distributed (i.i.d.) according to the distribution $p(x)$ that they should represent. If $p(x)$ is difficult to sample, one can sample from some other feasible importance density $g(x)$ while correcting the introduced bias by sample weighting. A set of weighted particles $\{\langle x_i, \pi_i \rangle\}$ with $x_i \sim g(x)$ approximates then an arbitrary distribution p according to

$$\Pr(\mathcal{A}) = \int_{\mathcal{A}} p(x)\, dx \approx \sum_i \pi_i\, \delta_{\mathcal{A}}(x_i), \quad \pi_i = \frac{p(x_i)}{g(x_i)}$$

with $\delta_{\mathcal{A}}$ denoting the characteristic function of an arbitrary set $\mathcal{A}$. Given a weighted particle representation $\{\langle \bar{x}_i, \bar{\pi}_i \rangle\}$ for the belief at time $t-1$, Bayes recursion (Eq. 1-2) becomes, modulo a normalization factor,

$$p(x_t|z_{1:t}) \approx\ p(z_t|x_t) \sum_i \bar{\pi}_i\, p(x_t|\bar{x}_i). \tag{41}$$

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

$$g(x_t) = \sum_i \pi_i\, p(x_t|x_i). \tag{42}$$

At each iteration t, a new set of representative particles, $\{x_i\}$, is i.i.d.-sampled from $g(x)$. Then, observation z_t is analyzed for new importance weights $\pi_i = p(z_t|x_i)\bar{\pi}_i$. To focus on likely hypotheses, particles are periodically resampled according to their weights [34]–[36].

B. HJS particle filtering

The particle filter implementation of the HJS tracker is outlined in Algorithm 1. Each marginal belief is represented via N weighted samples, $\{\langle \bar{x}_i^k, \bar{\pi}_i^k \rangle\}$.

1) Prediction with forward MRF model: If the model in Eq. 16 is Gaussian and initial messages are uniform or Gaussian, Eq. 18 defines a Gaussian mixture. Even under these restrictions direct sampling is infeasible since the mixture size increases exponentially with the number of targets and the number of BP iterations. Manageable solutions have been proposed: PAMPAS [23] is an effective algorithm if the MRF is Gaussian; Nonparametric Belief Propagation (NBP) [24] can be used with generic models and produces (biased) kernel estimates from which samples can be drawn. We propose an alternative, more efficient solution which is suitable for tracking applications. First, a particle representation $\{x_i^k\}$ of $p^-(x_t^k|z_{1:t-1})$ is assembled by sampling N times from the importance mixture in Eq. 42 [36]. These samples are supposed to be representative for the marginals as well, so that BP can be run to iteratively adjust their importance weights $\{\pi_i^k\}$ (Eq. 17, 18). This assumption is acceptable if $p^-(x_t^k|z_{1:t-1})$ and $p(x_t^k|z_{1:t-1})$ are close, a requirement that is usually met by tracking applications when MRF potentials are smooth radial functions with small support. Given these restrictions, this method is faster than PAMPAS or NBP since no further sampling is required and all values needed for message updating can be pre-computed and accessed via lookup table. With $\bar{m}_i^{h,k}, m_i^{h,k}$ denoting previous and current messages evaluated at particle x_i^k, convergence is detected if KL-divergence of subsequent beliefs falls below a threshold θ_{BP}:

$$\sum_i \sum_{h \in \mathcal{C}_k} \ln \frac{m_i^{h,k}}{\bar{m}_i^{h,k}} \prod_{h \in \mathcal{C}_k} m_i^{h,k} < \theta_{\text{BP}} \tag{43}$$

In Sec. VI-C we show that when tracking 5 people with spatial exclusion MRF and $\theta_{\text{BP}} = 0.01$, at most 5 iterations are required. This overhead is negligible, and adjustment of $\mathcal{E}$ is not needed (we assume always fully connected graph).

2) Update with HJS likelihood: Two image buffers are assigned to each target, b_{fg}^k and b_{bg}^k, which are used as support to incrementally calculate occlusion maps and occlusion evidence. All

Algorithm 1 HJS particle filter iteration

input: $\{\langle \bar{x}_i^k, \bar{\pi}_i^k \rangle\}$ are particle sets representing $p(x_{t-1}^k | z_{1:t-1})$

predict:

foreach object index k **do**

resample N particle indices i.i.d. according to $\{\bar{\pi}_i^k\}$

foreach selected particle index n **do**

sample new particle x_i^k from $p(x_t^k | \bar{x}_n^k)$, set π_i^k to 1

compute all $p_{i,j}^{h,k} = p(x_i^k, x_j^h)$, initialize $\bar{m}_i^{k,h} = 1$, $\mathcal{D}_{\text{KL}} > \theta_{\text{BP}}$

while $\mathcal{D}_{\text{KL}} > \theta_{\text{BP}}$ **do**

compute new messages $m_i^{h,k}$ according to Eq. 17

normalize messages such that $\sum_i m_i^{h,k} = 1$

update KL–divergence $\mathcal{D}_{\text{KL}}$ according to Eq. 43

compute all particle weights π_i^k according to Eq. 18

$\{\langle x_i^k, \pi_i^k \rangle\}$ are particle sets representing $p(x_t^k | z_{1:t-1})$

update:

order all particles $\{x_i^k\}$ according to camera distance

let $\{x_p, k_p\}$ be the ordered, hybrid, particle set

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

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

compute occlusion map $w_{x_p}^* = \Delta_g(x_p) \prod_{h \neq k_p} (b_{\text{fg}}^h / N)$

compute foreground term fc_p (Eq. 33)

compute background term bc_p (Eq. 40)

assign marginal likelihood $\pi_p = \exp(\frac{\text{fc}_p + \text{bc}_p}{2\sigma^2})$

update fg buffer $b_{\text{fg}}^{k_p} = b_{\text{fg}}^{k_p} - \Delta_g(x_p)$

foreach object index $h \neq k_p$ **do**

compute reduced map $\hat{w}_{x_p}^* = \prod_{m \neq h, k_p} (b_{\text{fg}}^m / N)$

compute reduced foreground term rfc_p

update bg buffer $b_{\text{bg}}^{k_p} = b_{\text{bg}}^{k_p} + \text{rfc}_p \hat{w}_{x_p}^* \Delta_g(x_p)$

output: $\{\langle x_i^k, \pi_i^k \rangle\}$ are particle sets representing $p(x_t^k | z_{1:t})$

mathematical operations on these buffers ($+$, $-$, $\cdot$) are intended to be applied at each single pixel. The procedure visits particles $\{x_p\}$ from the closest to the camera to the farthest one. First, the candidate occlusion map $w^*_{x_p}$ with support on its silhouette, $\Delta_g(x_p)$, is assembled pixel-wise as the product of other target foreground buffers according to Eq. 34, and then used to calculate the foreground term as by Eq. 33. The background term is extracted from the corresponding buffer $b^{k_p}_{\text{bg}}$ as by Eq. 40. After updating the candidate particle weight with its marginal likelihood, the foreground buffer belonging to the particle, $b^{k_p}_{\text{fg}}$, is updated by subtracting the candidate silhouette, $\Delta_g(x_p)$. Finally, the background buffer needs to be updated. A reduced map $\hat{w}^*_{x_p}$ is calculated by ignoring in turn each one of the other targets. It is then used to reevaluate Eq. 33 on the candidate state x_p. The candidate silhouette is then accumulated into the background buffer $b^{k_p}_{\text{bg}}$ with intensity according to Eq. 39. As for the single line-of-sight in [1], the complexity of the measurement update is at most quadratic in the number of tracked targets K and linear in the number of particles N. As far as image resolution is concerned, the algorithm scales linearly with the average silhouette size S making the overall update complexity $\mathcal{O}(K^2NS)$.

VI. Experiments

A. Experimental setup

Experiments have been carried out on real video captured at 15 Hz by a calibrated camera monitoring a small room. Target state is defined as position and velocity on the floor plane, thus we have 4D state spaces. A linear-Gaussian dynamical model with variance 0.6m/s is assumed for each target. A spatial exclusion MRF is implemented: $p(x^h, x^k) = \delta(\|x^h - x^k\| > 40\text{cm})$. Target appearance is described by histogram triples (RGB, 8 bins per channel, head/torso/legs) acquired for front/side/back views, and generalized-cylinder shape model [7]. Model histograms are acquired prior to tracking as in Fig. 5. Candidate histograms are scored against a model rendered as follows: the two model views closest to the candidates' relative orientation (i.e. direction of velocity component) are identified and their histograms accordingly interpolated. 1-body log-likelihood is computed as the normalized sum of body part contributions calculated according to Eq. 26, with d implementing Bhattacharyya-coefficient based distance [33]. Using three histograms (instead of only one for the whole body) increases its sensitivity since body parts are usually visually uncorrelated. Likelihood width σ (Eq. 19), occlusion penalty d_{occ} (Eq. 26) and number of particles N are specified below. All tracks are initialized by hand, image resolution

Fig. 5. Appearance model acquisition. Three pictures are taken for each target, sampling a front, side and back view. Foreground blobs are extracted by background suppression and morphological noise cleaning. Head position is estimated by triangulation, and target height is derived. Projected shape identifies blob parts from which head-torso-legs histograms are extracted.

is down-sampled to 200×150 pixels. To underline robustness of the approach, no strategy to recover from failures is implemented (as e.g. used by BraMBLe [7]).

B. HJS tracking performance

Three sequences of graded difficulty have been processed: three, four and five people were moving randomly in the camera's field of view at the same time, leading to severe occlusions, also of considerable duration. The challenges posed to a tracker by these sequences can best be observed in Fig. 8. Pseudo-ground truth has been acquired for all videos as follows. Three other cameras installed in the remaining corners of the room were capturing the scene while the sequences have been acquired. A single movie is created for each sequence by assembling all images captured by these three cameras (this movie does not contain any frame utilized later on for evaluation). We ran the tracker with $N = 500$ particles per target on these multi-view data, stored the particles at each frame and post-processed them by running the Viterbi algorithm to find maximum likelihood trajectories using kernel density estimates. We finally validated the positions by visual inspection; quality of pseudo-ground truth is about the precision a human labeler would be able to obtain. This data is used for performance analysis: the graphs in Fig. 6 show the fraction of resampled particles that fall within a distance from the true target position, plotted against this distance. This is the particle approximation of the posterior mass distributed around the true position.

The first sequence (*seq1*, 3 people, 670 frames) has been processed with a joint, separable and HJS particle filter using different numbers of particles. Likelihood parameters are $\sigma = 0.12$, $d_{\text{occ}} = 0.25$. Distribution of localization error for the different cases is shown in Fig. 6, left plot. The separable filter is not able to track this sequence, neither with $N = 100$ (S_100p) nor with $N = 500$ (S_500p) particles per target. The only effect of increasing the number of particles

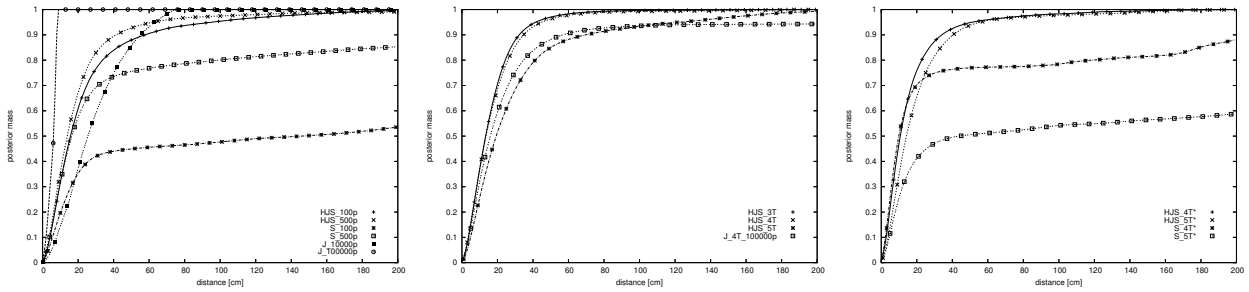

Fig. 6. Distribution of posterior mass around the ground truth, plotted as a function of localization error. **Left plot**, *seq1*: HJS filter with 100 particles (HJS_100p), HJS filter with 500 particles (HJS_500p), separable filter with 100 particles (S_100p), separable filter with 500 particles (S_500p), joint filter with 10000 particles (J_10000p), joint filter 100000 particles (J_100000p). **Middle plot**, 100 particles: HJS filter on *seq1* (HJS_3T), HJS filter on *seq2* (HJS_4T), HJS filter on *seq3* (HJS_5T), joint filter on *seq2* with 100000 particles (J_4T_100000p). **Right plot**, real time processing using two cameras, 100 particles: HJS filter on *seq2* (HJS_4T*), HJS filter on *seq3* (HJS_5T*), separable filter on *seq2* (S_4T*), separable filter on *seq3* (S_5T*).

is to postpone the failure. The joint filter behaves well as expected, but needs a huge number of particles: with $N = 10^4$ joint particles (J_10000p, needs 4sec per frame) it does not fail but accuracy is not exciting due to sparse sampling of joint state space; increasing N by an order of magnitude ($N = 10^5$, J_100000p) gives excellent precision but needs about 80sec to process a frame. HJS filter with $N = 100$ particles per target (HJS_100p) does keep accurate track over the whole sequence (more than 85% of the particles fall within 50cm from the ground truth), and it is easily accommodated in real time on a 3GHz PC (see Table I). Increasing the representation size to $N = 500$ does only slightly improve accuracy, and slows down processing by a factor of 5 (as expected; recall that complexity is $\mathcal{O}(K^2N)$). Not reported experiments suggest that going below 80 particles becomes critical. Note that in order to consistently represent involved posteriors, N must be adapted to intrinsic uncertainty which varies with the complexity of a sequence (e.g. permanence within and extent of occlusion volumes). Empirically, $N = 100$ seems to be sufficient for the sequences used in this section. Online adaptation of sample size will be subject of future research. To experimentally justify the choice of likelihood proposed in Eq. 26 we ran the joint filter with $N = 10^5$ using a conventional form of likelihood, i.e. that accounts only for visible parts of the targets (no occlusion penalty is assigned). Images in Fig. 7 show how *phantom particles* were generated which finally led to track loss.

The second plot in Fig. 6 shows how the HJS filter behaves with increasing task complexity. Accuracy of tracking using $N = 100$ particles per target is shown on *seq1*, *seq2* (4 people,

Fig. 7. Joint filter on *seq1*, 10^5 particles, likelihood without occlusion penalty. Shown are frames 1, 16, 33.

TABLE I

REAL TIME PERFORMANCE ON A 3GHZ PC.

targets	CPU load	frame rate
3	50-80%	15Hz
4	90-100%	7-15Hz
5	100%	3-10Hz

618 frames) and *seq3* (5 people, 341 frames). Likelihood parameters have been set to $\sigma = 0.1$, $d_{\text{occ}} = 0.15$, and are adopted for all remaining experiments. This explains why the curve reported here for *seq1* shows higher accuracy than in the previous case. More pronounced likelihood responses provide sharper occlusion maps which lead to better estimates under occlusion. The parameters have been selected empirically; a principled rule may be found by building on the work of [15]. All three sequences have been successfully tracked by the HJS filter. Accuracy diminishes with increasing number of targets because of more frequent occlusions. Note that when a body is completely occluded there is no information to localize it exactly, its position can only be bound to the estimated occlusion volume. This is not taken into account by the plots which always consider particle distance from the true position. To compare the results obtained we have processed *seq2* (4 targets) with the joint tracker using 10^5 joint particles. Due to sparse sampling of joint space it fails. Consider that 100 HJS particles per target represent 10^8 joint particles.

Finally, we investigated real time performance on a 3GHz PC. CPU load and processed frame rate for *seq1-2-3* are reported in Table I, using 100 particles per target, on image resolution of 200×150 pixels. Frame rate varies because of reduced computations when no occlusion is detected (HJS filter propagates a DBN, Sec. IV-B.3). With four targets (*seq2*) the frame rate goes not below 7Hz, which enables real time processing. Frame rate with five targets (*seq3*) becomes

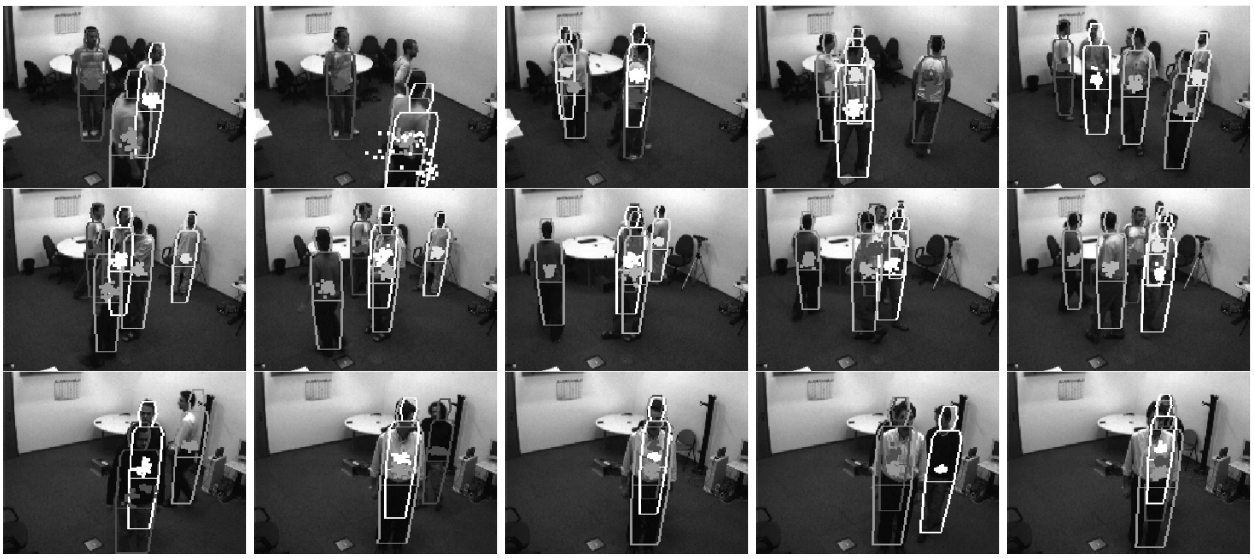

Fig. 8. Key frames from evaluation: *seq1*, separable filter, frames 56, 62; *seq2*, HJS filter, frames 198, 230; *seq3*, HJS filter, frames 28, 59, 71, 93, 109, 119; *seq4*, HJS filter, frames 34, 66, 134, 147, 172, 194. Estimates are displayed in different colors; generalized-cylinder model is rendered at the posterior mean.

critical under multiple occlusions (3Hz). On this sequence we ran the filter online using two cameras providing orthogonal views of the scene. Images were still elaborated sequentially, by processing one frame at a time and alternating between views. This has the effect of reducing uncertainty since occlusions are less frequent. We made several trials: 7 times out of 10 the HJS filter could successfully track the whole sequence; statistics on a successful run is reported in Fig. 6. The faster separable filter (15Hz) cannot track, neither *seq2* nor *seq3*.

C. Preventing coalescence with spatial exclusion MRF

The gain achieved with the spatial exclusion MRF is analyzed on a short sequence (*seq4*, 215 frames) with three targets, two of them sharing almost identical models (there is a slight difference in their height). Some key frames are shown in Fig. 8. Both independent 1-body trackers and HJS tracker without MRF coalesce around frame 30, when the first occlusion takes place. Spatial exclusion introduces a sort of repulsive field at each mode, which prevents the particles of the occluded target from invading the space occupied by the (well localized) occluding object. Very challenging is the situation around frames 140-200, where these targets become completely occluded by a third one. HJS likelihood bounds their particles to the occlusion

cone generated by the closest target, which remain well separated due to the MRF. Note also that since the occlusion volume is narrow and MRF avoids collisions, there is not enough space for the particle sets to swap their order. Given this, the tracker agilely associates the correct track to the correct target when it leaves the occlusion, tracks it while it changes its distance from the camera, and reassigns the correct order when it enters the occlusion volume again. We have performed several trials: sometimes the tracker underestimates the distance of the target that becomes again visible so that when it reenters the occlusion, the order is switched. However, the tracker never failed because of an occlusion.

VII. CONCLUSIONS

A theory and implementation of a Bayesian approach to multibody tracking has been presented. While previous work has addressed this problem by explicitly propagating the joint statistics over time, this method maintains a compact representation in form of product of marginals which is shown to be optimal in terms of joint reconstruction from single target distributions. The update of such a representation is made efficient while still using non-separable form of likelihood, derived from imaging principles. An MRF correction to independently propagated 1-body priors accounts for simple interactions such as spatial exclusion. The resulting method uses a representation size that grows linearly with the number of targets, and a computational complexity that grows quadratically. A particle filter is developed which provides an efficient engine enabling the use of rich appearance descriptors able to distinguish between different targets, a capability that is prevented by the computational burden of traditional methods. The method runs robustly in real time with four to five targets.

Future work aims at investigating adaptation of representation size to the performance of the tracker. Such a strategy would self-manage the efficiency-robustness trade-off, and would tend to populate the state space regions that map to other objects occlusion volumes when a target is not detected anywhere else. The use of lower-dimensional, thus less expensive, likelihoods and their impact on accuracy and robustness, as well as the introduction of a background model will also be investigated. The update strategy is indeed not limited to histogram based models, but may be applied to measurement vectors extracted with more generic image filters.

ACKNOWLEDGMENT

The author would like to thank Roberto Brunelli (ITC-irst) for helpful discussions and comments, and Roberto Manduchi (UCSC) for jointly working on a predecessor of this work [1]. Sincere thanks go to the anonymous reviewers who provided helpful and constructive comments. The author has been partly funded by Provincia Autonoma di Trento under project PEACH (Personalized Experience of Active Cultural Heritage), and by the European Union under project CHIL (Computers in the Human Interaction Loop, IP 506909).

REFERENCES

[1] O. Lanz and R. Manduchi, "Hybrid joint-separable multibody tracking," in *Int. Conf. Computer Vision and Pattern Recognition*, 2005.

[2] J. MacCormick and A. Blake, "Probabilistic exclusion and partitioned sampling for tracking multiple objects," *Int. Journal of Computer Vision*, vol. 39, no. 1, 2000.

[3] D. Tweed and A. Calway, "Tracking many objects using subordinated CONDENSATION," in *British Machine Vision Conference*, 2002.

[4] H. Tao, H. S. Sawhney, and R. Kumar, "A sampling algorithm for detecting and tracking multiple objects," in *Vision Algorithms Workshop*, 1999.

[5] 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*, 2004.

[6] B. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for Bayesian multi-target filtering with Random Finite Sets," *IEEE Trans. Aerospace and Electronic Systems*, vol. 41, no. 4, 2005.

[7] M. Isard and J. MacCormick, "BraMBLe: A bayesian multiple-blob tracker," in *Int. Conf. Computer Vision*, 2003.

[8] J. Sullivan, A. Blake, M. Isard, and J. MacCormick, "Bayesian object localisation in images," *Int. Journal of Computer Vision*, vol. 44, no. 2, 2001.

[9] J. Vermaak, A. Doucet, and P. Perez, "Maintaining multi-modality through mixture tracking," in *Int. Conf. Computer Vision*, 2001.

[10] Z. Khan, T. Balch, and F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, 2005.

[11] T. Yu and Y. Wu, "Collaborative tracking of multiple targets," in *Int. Conf. Computer Vision and Pattern Recognition*, 2005.

[12] J. Deutscher, A. Blake, and I. Reid, "Articulated body motion capture by annealed particle filtering," in *Int. Conf. Computer Vision and Pattern Recognition*, 2000.

[13] C. Sminchisescu and B. Triggs, "Fast mixing hyperdynamic sampling," *Journal of Image and Vision Computing, Special Issue on Outstanding Papers from ECCV 2002*, 2006 (to appear).

[14] K. Choo and D. Fleet, "People tracking with Hybrid Monte Carlo," in *Int. Conf. Computer Vision*, 2001.

[15] J. Sullivan and J. Rittscher, "Guiding random particles by deterministic search," in *Int. Conf. Computer Vision*, 2001.

[16] C. Sminchisescu and B. Triggs, "A robust multiple hypothesis approach to monocular human motion tracking," INRIA, Tech. Rep. 4208, 2001.

[17] M. Cover and J. Thomas, *Elements of Information Theory*. Wiley, 1991.

[18] J. Coughlan and S. Ferreira, "Finding deformable shapes using loopy belief propagation," in *European Conf. Computer Vision*, 2002.

[19] J. Sun, H. Shum, and N. Zheng, "Stereo matching using belief propagation," in *European Conf. Computer Vision*, 2002.

[20] L. Sigal, S. Bhatia, S. Roth, M. Black, and M. Isard, "Tracking loose-limbed people," in *Int. Conf. Computer Vision and Pattern Recognition*, 2004.

[21] E. Sudderth, M. Mandel, W. Freeman, and A. Willsky, "Distributed occlusion reasoning for tracking with nonparametric belief propagation," in *Advances in Neural Information Processing Systems*. MIT Press, 2005.

[22] J. Yedidia, W. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," in *Exploring Artificial Intelligence in the New Millennium*. Morgan Kaufmann Publishers Inc., 2003.

[23] M. Isard, "PAMPAS: Real-valued graphical models for computer vision," in *Int. Conf. Computer Vision and Pattern Recognition*, 2003.

[24] E. Sudderth, A. Ihler, W. Freeman, and A. Willsky, "Nonparametric belief propagation," in *Int. Conf. Computer Vision and Pattern Recognition*, 2003.

[25] K. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, UC Berkeley, 2002.

[26] Q. Diao, J. Lu, W. Hu, Y. Zhang, and G. Bradski, "DBN models and a prediction method for visual tracking," in *Bayesian Modeling Applications Workshop*, 2003.

[27] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, 1999.

[28] M. Opper and D. Saad, *Advanced Mean Field Methods*. MIT Press, 2001.

[29] S. Li, *Markov Random Field Modeling in Computer Vision*. Springer, 1995.

[30] A. Santuari, O. Lanz, and R. Brunelli, "Synthetic movies for computer vision applications," in *Int. Conf. Visualization, Imaging, and Image Processing*, 2003.

[31] H. Sidenbladh and M. Black, "Learning the statistics of people in images and video," *Int. Journal of Computer Vision*, vol. 54, no. 1, 2003.

[32] A. Jepson, D. Fleet, and T. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, 2003.

[33] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, 2003.

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

[35] M. Isard and A. Blake, "CONDENSATION – conditional density propagation for visual tracking," *Int. Journal of Computer Vision*, vol. 29, no. 1, 1998.

[36] 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*, vol. 50, no. 2, 2002.

Oswald Lanz received the laurea degree in Mathematics and the PhD degree in Information and Communication Technologies from the University of Trento in 2000 and 2005, respectively. He then joined the Istituto Trentino di Cultura (ITC-irst), where he is currently a research fellow in the Technology of Vision (TeV) group. His current research interests are about Computer Vision, including probabilistic inference and learning, sensor cooperation and fusion, and adaptive aspects of monitoring systems.