Preprint: Pattern Recognition, Vol. 90, No. 6, pp. 377-389, 2019.

A Labeled Random Finite Set Online Multi-Object Tracker for Video Data

Du Yong Kim[a], Ba-Ngu Vo[b], Ba-Tuong Vo[b], Moongu Jeon[c]

[a]*School of Engineering, RMIT University, Melbourne, Australia*
[b]*Department of Electrical and Computer Engineering, Curtin University, Bentley, Australia*
[c]*School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, Korea*

Abstract

This paper proposes an online multi-object tracking algorithm for image observations using a top-down Bayesian formulation that seamlessly integrates state estimation, track management, handling of false positives, false negatives and occlusion into a single recursion. This is achieved by modeling the multi-object state as labeled random finite set and using the Bayes recursion to propagate the multi-object filtering density forward in time. The proposed filter updates tracks with detections but switches to image data when detection loss occurs, thereby exploiting the efficiency of detection data and the accuracy of image data. Furthermore the labeled random finite set framework enables the incorporation of prior knowledge that detection loss in the middle of the scene are likely to be due to occlusions. Such prior knowledge can be exploited to improve occlusion handling, especially long occlusions that can lead to premature track termination in on-line multi-object tracking. Tracking performance is compared to state-of-the-art algorithms on synthetic data and well-known benchmark video datasets.

Keywords: online multi-object tracking, Track-before-detect, random finite set

1. Introduction

In a multiple object setting, not only do the states of the objects vary with time, but the number of objects also changes due to objects appearing and disappearing. In this work, we consider the problem of jointly estimating the time-varying number of objects and their trajectories from a stream of noisy images. In particular, we are interested in multi-object tracking (MOT) solutions that compute estimates at a given time using only data up to that time. These so-called online solutions are better suited for time-critical applications.

A critical function of a multi-object tracker is track management, which concerns track initiation/termination and track labeling or identifying trajectories of individual objects. Track management is more challenging for online algorithms than for batch algorithms. Usually, track initiation/termination in online MOT algorithms is performed by examining consecutive detections in time [1], [2]. However, false positives generated by the background, compounded by false negatives (including those from object occlusions), can result in false tracks and lost tracks, especially in online algorithms. False negatives also cause track fragmentation in batch algorithms as reported in [3], [4], [5] [6]. With the exception of the recent network flow [7] techniques, track labels are assigned upon track initiation, and maintained over time until termination. An online multi-object Bayesian filter that provides systematic track labeling using labeled random finite set (RFS) was proposed in [8].

In most video MOT approaches, each image in the data sequence is compressed into a set of detections before a *filtering* operation is applied to keep track of the objects (including undetected ones). Typically, in the filtering module, motion correspondence or data association is first determined followed by the application of standard filtering techniques such as Kalman or sequential Monte Carlo [1, 2]. The main advantage of performing detection before filtering is the computational efficiency in the compression of images into relevant detections. The main disadvantage is the loss of information, in addition to false negatives and false positives, especially in low signal to noise ratio (SNR) applications.

Track-before-detect (TBD) is an alternative approach, which by-passes the detection module and exploits the spatio-temporal information directly from the image sequence. The TBD methodology is often required in tracking applications for low SNR image data [9], [10], [11], [12]. In visual tracking applications, perhaps the most well-known TBD MOT algorithm is BraMBLe [13]. Other visual MOT algorithms that can be categorized as TBD include [14], [15] which exploit color-based observation models, [16], [2], which exploit multi-modality of distributions, and [17] which uses multi-Bernoulli random finite set models. While the TBD approach minimizes information loss, it is computationally more expensive. So far it is not clear how we could simultaneously process detection and image measurements to exploit their complementary advantages, in a principled manner.

In this paper, we develop an efficient online MOT algorithm for video data that exploits the advantages of both detection-based and TBD approaches to improve performance while reducing the computational cost. In the visual MOT literature, simultaneous consideration of detections and image features were proposed in ad-hoc manners [1], [5], and it is not clear how to combine them in a principled way. The innovation of our proposed algorithm is the adaptive update of tracks with detections (for efficiency), or with local regions of the input

image (to minimize information loss and improve accuracy). In addition, the proposed visual MOT filter seamlessly integrates state estimation, track management, clutter rejection, false negatives and occlusion handling, (which are traditionally separate functionalities) in a single Bayesian recursion.

The key technical contribution is a hybrid multi-object measurement model that simultaneously accommodates detections and image observations. Conceptually, this model is a simple generalization of the standard multi-object measurement model [18] and the separable model for image measurement [10]. Such a simple construct, however, enables us to simultaneously exploit the efficiency of the detection-based approach and the accuracy of TBD-based approach. Specifically, using the labeled RFS framework for multi-object estimation [8], we prove conjugacy of the Generalized Labelled Multi-Bernoulli (GLMB) distributions with respect to the likelihood function of the proposed measurement model. Using this conjugacy result, and the labeled RFS estimation formulation [8], we develop an analytic Bayesian MOT filter that avoids processing the entire image so as to reduce computational costs, while at the same time make use of relevant local information at the image level to reduce the effect of false negatives as well as tracking errors.

Due to the labeled RFS filtering formulation, the proposed MOT filter addresses state estimation, track management, clutter rejection, false negatives and occlusion handling, in one single recursion. Generally, an online MOT algorithm would terminate a track that has not been detected over several frames. In many visual MOT applications however, it is observed that away from designated exit regions such as scene edges, the longer an object is in the scene, the less likely it is to disappear, see for example [19], [20] which exploit theses so-called closed world assumptions. Intuitively, this observation can be used to delay the termination of tracks that have been occluded over an extended period, so as to improve occlusion handling. The labeled RFS framework provides a principled and inexpensive means to exploit this observation for improved occlusion handling.

The remainder of the paper is structured as follows. The Bayesian filtering formulation of the MOT problem using labeled RFS is given in Section 2, followed by details of the proposed solution in Section 3. Performance evaluation of the proposed MOT filter against state-of-the-art trackers is presented in Section 4, and concluding remarks are given in Section 5.

2. Bayesian Multiple Object Tracking

This section outlines the RFS framework for MOT that accommodates uncertainty in the number of objects, the states of the objects and their trajectories. The salient feature of this framework is that it admits direct parallels between traditional Bayesian filtering and MOT. The modeling of the multi-object state as an RFS in Subsection 2.1 enables Bayesian filtering concepts to be directly translated to the multi-object case in Subsection 2.2. Subsection 2.3 examines the MOT problem in the presence of occlusion.

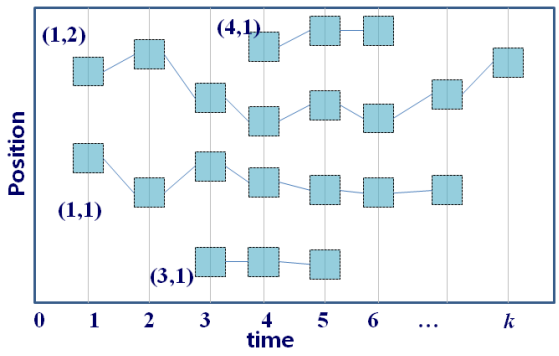

Figure 1: 1D multi-object trajectories with labeling

2.1. Multi-object State

To distinguish different object trajectories in a multi-object setting, each object is assigned a unique label ℓ_k that consists of an ordered pair (t, i), where t is the time of birth and i is the index of individual objects born at the same time [8]. For example, if two objects appear in the scene at time 1, one is assigned label (1,1) while the other is assigned label (1,2), see Figure 1. A trajectory or track is the sequence of states with the same label.

Formally, the state of an object at time k is a vector $\mathbf{x}_k = (x_k, \ell_k) \in \mathbb{X} \times \mathbb{L}_k$, where $\mathbb{L}_k$ denotes the label space for objects at time k (including those born prior to k). Note that $\mathbb{L}_k$ is given by $\mathbb{B}_k \cup \mathbb{L}_{k-1}$, where $\mathbb{B}_k$ denotes the label space for objects born at time k (and is disjoint from $\mathbb{L}_{k-1}$). Suppose that there are N_k objects at time k, with states $\mathbf{x}_{k,1}, ..., \mathbf{x}_{k,N_k}$. In the context of MOT, the collection of states, referred to as the *multi-object state*, is naturally represented as a finite set

$$\mathbf{X}_k = \{\mathbf{x}_{k,1}, ..., \mathbf{x}_{k,N_k}\} \in \mathcal{F}(\mathbb{X} \times \mathbb{L}_k),$$

where $\mathcal{F}(\mathbb{X} \times \mathbb{L}_k)$ denotes the space of finite subsets of $\mathbb{X} \times \mathbb{L}_k$. We denote cardinality (number of elements) of $\mathbf{X}$ by $|\mathbf{X}|$ and the set of labels of $\mathbf{X}$, $\{\ell : (x, \ell) \in \mathbf{X}\}$, by $\mathcal{L}(\mathbf{X})$. Note that since the label is unique, no two objects have the same label, i.e. $\delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|) = 1$. Hence $\Delta(\mathbf{X}) \triangleq \delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|)$ is called the *distinct label indicator*.

For the rest of the paper, we follow the convention that single-object states are represented by lower-case letters (e.g. x, $\mathbf{x}$), while multi-object states are represented by upper-case letters (e.g. X, $\mathbf{X}$), symbols for labeled states and their distributions are bold-faced to distinguish them from unlabeled ones (e.g. $\mathbf{x}$, $\mathbf{X}$, π, etc.), and spaces are represented by blackboard bold (e.g. $\mathbb{X}$, $\mathbb{Z}$, $\mathbb{L}$, $\mathbb{N}$, etc.). The list of variables $X_m, X_{m+1}, ..., X_n$ is abbreviated as $X_{m:n}$. We denote a generalization of the Kroneker delta that takes arbitrary arguments such as sets, vectors, integers etc., by

$$\delta_Y[X] \triangleq \begin{cases} 1, & \text{if } X = Y \\ 0, & \text{otherwise} \end{cases}.$$

For a given set S, $1_S(\cdot)$ denotes the indicator function of S, and $\mathcal{F}(S)$ denotes the class of finite subsets of S. For a finite set

X, the multi-object exponential notation f^X denotes the product $\prod_{x\in X} f(x)$, with $f^{\emptyset} = 1$. The inner product $\int f(x)g(x)dx$ is denoted by $\langle f, g\rangle$.

2.2. Multi-object Bayes filter

From a Bayesian estimation viewpoint the multi-object state is naturally modeled as an RFS or a simple-finite point process [21]. While the space $\mathcal{F}(\mathbb{X}\times\mathbb{L}_k)$ does not inherit the Euclidean notion of probability density, Mahler's Finite Set Statistic (FISST) provides a suitable notion of integration/density for RFSs [18, 22]. This approach is mathematically consistent with measure theoretic integration/density but circumvents measure theoretic constructs [23].

At time k, the multi-object state $\mathbf{X}_k$ is observed as an image y_k. All information on the set of object trajectories conditioned on the observation history $y_{1:k}$, is captured in the *multi-object posterior density*

$$\boldsymbol{\pi}_{0:k}(\mathbf{X}_{0:k}|y_{1:k}) \propto \prod_{j=1}^{k} g_j(y_j|\mathbf{X}_j)\mathbf{f}_{j|j-1}(\mathbf{X}_j|\mathbf{X}_{j-1})\boldsymbol{\pi}_0(\mathbf{X}_0)$$

where $\boldsymbol{\pi}_0$ is the initial prior, $g_j(\cdot|\cdot)$ is the *multi-object likelihood function* at time j, $\mathbf{f}_{j|j-1}(\cdot|\cdot)$ is the *multi-object transition density* to time j. The multi-object likelihood function encapsulates the underlying observation model while the multi-object transition density encapsulates the underlying models for motions, births and deaths of objects. Note that track management is incorporated into the Bayes recursion via the multi-object state with distinct labels.

MCMC approximations of the posterior density have been proposed in [24, 25] for detection measurements and image measurements respectively. Results on satellite imaging applications reported in [25] are very impressive. However, these techniques are still expensive and not suitable for on-line application.

For real-time tracking, a more tractable alternative is the *multi-object filtering density*, a marginal of the multi-object posterior. For notational compactness, herein we omit the dependence on data in the multi-object densities. The multi-object filtering density can be recursively propagated by the *multi-object Bayes filter* [21], [18] according to the following prediction and update

$$\boldsymbol{\pi}_{k+1|k}(\mathbf{X}_{k+1}) = \int \mathbf{f}_{k+1|k}(\mathbf{X}_{k+1}|\mathbf{X}_k)\boldsymbol{\pi}_k(\mathbf{X}_k)\delta\mathbf{X}_k, \tag{1}$$

$$\boldsymbol{\pi}_{k+1}(\mathbf{X}_{k+1}) = \frac{g_{k+1}(y_{k+1}|\mathbf{X}_{k+1})\boldsymbol{\pi}_{k+1|k}(\mathbf{X}_{k+1})}{\int g_{k+1}(y_{k+1}|\mathbf{X})\boldsymbol{\pi}_{k+1|k}(\mathbf{X})\delta\mathbf{X}}, \tag{2}$$

where the integral is a *set integral* defined for any function $\mathbf{f} : \mathcal{F}(\mathbb{X}\times\mathbb{L}_k) \to \mathbb{R}$ by

$$\int \mathbf{f}(\mathbf{X})\delta\mathbf{X} = \sum_{i=0}^{\infty} \frac{1}{i!} \int \mathbf{f}(\{\mathbf{x}_1, ..., \mathbf{x}_i\})d(\mathbf{x}_1, ..., \mathbf{x}_i).$$

Bayes optimal multi-object estimators can be formulated by minimizing the Bayes risk with ordinary integrals replaced by set integrals as in [22]. One such estimator is the marginal multi-object estimator [18].

A generic particle implementation of the Bayes multi-object filter Eq. (1)-(2) was proposed in [23] and applied to labeled multi-object states in [11]. The *Generalized labeled Multi-Bernoulli* (GLMB) filter is an analytic solution to the Bayes multi-object filter, under the standard multi-object dynamic and observation models [8].

2.2.1. Standard multi-object dynamic model

Given the multi-object state $\mathbf{X}_k$ (at time k), each state $(x_k, \ell_k) \in \mathbf{X}_k$ either survives with probability $P_{S,k}(x_k, \ell_k)$ and evolves to a new state (x_{k+1}, ℓ_{k+1}) (at time $k+1$) with probability density $f_{k+1|k}(x_{k+1}|x_k, \ell_k)\delta_{\ell_k}[\ell_{k+1}]$ or dies with probability $1 - P_{S,k}(x_k, \ell_k)$. The set $\mathbf{B}_{k+1}$ of new objects born at time $k+1$ is distributed according to the labeled multi-Bernoulli (LMB)

$$\Delta(\mathbf{B}_{k+1})\omega_{B,k+1}(\mathcal{L}(\mathbf{B}_{k+1}))p_{B,k+1}^{\mathbf{B}_{k+1}},$$

where $\omega_{B,k+1}(L) = \left[1_{\mathbb{B}_{k+1}} r_{B,k+1}\right]^L \left[1 - r_{B,k+1}\right]^{\mathbb{B}_{k+1}-L}$, $r_{B,k+1}(\ell)$ is the probability that a new object with label ℓ is born, and $p_{B,k+1}(\cdot, \ell)$ is the distribution of its kinematic state [8]. The multi-object state $\mathbf{X}_{k+1}$ (at time $k+1$) is the superposition of surviving objects and new born objects. It is assumed that, conditional on $\mathbf{X}_k$, objects move, appear and die independently of each other. The expression for the multi-object transition density $\mathbf{f}_{k+1|k}$ can be found in [8, 26]. The standard multi-object dynamic model enables the Bayes multi-object filter to address motion, births and deaths of objects.

2.2.2. Standard multi-object observation model

In most applications a designated detection operation D is applied to y_k resulting in a set of points

$$Z_k = D(y_k) \in \mathbb{Z}. \tag{3}$$

Since the detection process is not perfect, false positives and false negatives are inevitable. Hence only a subset of Z_k correspond to some objects in the scene (not all objects are detected) while the remainder are false positives. The most popular detection-based observation model is described in the following.

For a given multi-object state $\mathbf{X}_k$, each $(x, \ell) \in \mathbf{X}_k$ is either detected with probability $P_{D,k}(x, \ell)$ and generates a detection $z \in Z_k$ with likelihood $g_{D,k}(z|x, \ell)$ or missed with probability $1 - P_{D,k}(x, \ell)$. The *multi-object observation* Z_k is the superposition of the observations from detected objects and Poisson clutter with intensity κ_k. The ratio

$$\sigma_{D,k}(z|x, \ell) \triangleq \frac{g_{D,k}(z|x, \ell)}{\kappa_k(z)} \tag{4}$$

can be interpreted as the detection signal to noise ratio (SNR).

Assuming that, conditional on $\mathbf{X}_k$, detections are independent of each other and clutter, the multi-object likelihood function is given by [18], [8, 26]

$$g_k(y_k|\mathbf{X}_k) \propto \sum_{\theta\in\Theta_k(\mathcal{L}(\mathbf{X}_k))} \prod_{(x,\ell)\in\mathbf{X}_k} \psi_{D(y_k)}^{(\theta(\ell))}(x, \ell) \tag{5}$$

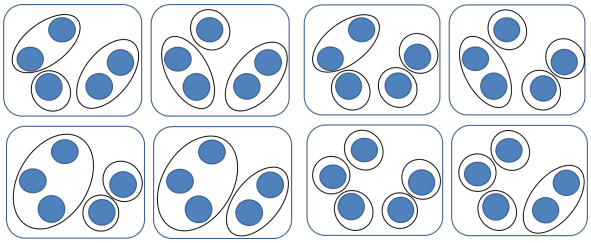

Figure 2: Examples of partitions for five objects

where: $\Theta_k(I)$ is the set of *positive 1-1* maps $\theta : I \to \{0{:}|Z_k|\}$, i.e. maps such that *no two distinct arguments are mapped to the same positive value*; and

$$\psi^{(j)}_{\{z_{1:M}\}}(x,\ell) = \begin{cases} P_{D,k}(x,\ell)\sigma_{D,k}(z_j|x,\ell), & \text{if } j = 1{:}M \\ 1 - P_{D,k}(x,\ell), & \text{if } j = 0 \end{cases}. \quad (6)$$

The map θ specifies which objects generated which detections, i.e. object ℓ generates detection $z_{\theta(\ell)} \in Z_k$, with undetected objects assigned to 0. The positive 1-1 property means that θ is 1-1 on $\{\ell : \theta(\ell) > 0\}$, the set of labels that are assigned positive values, and ensures that any detection in Z_k is assigned to at most one object.

The standard multi-object observation model enables the Bayes multi-object filter to address false negatives and false positives, but not occlusion. It assumes that each object is detected independently from each other, and that a detection cannot be assigned to more than one object. This is clearly not valid in occlusions.

2.3. Bayes Optimal Occlusion Handing

By relaxing the assumption that each object is independently detected, a multi-object observation model that explicitly addresses occlusion (as well as false negatives and false positives) was proposed in [27]. The main difference between this so-called *merged-measurement* model and the standard model is the idea that each group of objects (instead of each object) in the multi-object state generates at most one detection [27]. Figure 2 shows various partitions or groupings of a multi-object state with five objects.

A *partition* $\mathcal{U}_{\mathbf{X}}$ of a finite set $\mathbf{X}$ is a collection of mutually exclusive subsets of $\mathbf{X}$, whose union is $\mathbf{X}$. The collection of all partitions of $\mathbf{X}$ is denoted by $\mathcal{P}(\mathbf{X})$. It is assumed that given a partition $\mathcal{U}_{\mathbf{X}}$, each group $\mathbf{Y} \in \mathcal{U}_{\mathbf{X}}$ generates at most one detection with probability $P_{D,k}(\mathbf{Y})$, independent of other groups, and that conditional on detection generates z with likelihood $g_{D,k}(z|\mathbf{Y})$.

Let $\mathcal{L}(\mathcal{U}_{\mathbf{X}})$ denote the collection of labels of the partition $\mathcal{U}_{\mathbf{X}}$, i.e. $\mathcal{L}(\mathcal{U}_{\mathbf{X}}) \triangleq \{\mathcal{L}(\mathbf{Y}) : \mathbf{Y} \in \mathcal{U}_{\mathbf{X}}\}$ (note that $\mathcal{L}(\mathcal{U}_{\mathbf{X}})$ forms a partition of $\mathcal{L}(\mathbf{X})$). Let $\Xi_k(\mathcal{L}(\mathcal{U}_{\mathbf{X}}))$ denote the class of all positive 1-1 mappings $\vartheta : \mathcal{L}(\mathcal{U}_{\mathbf{X}}) \to \{0{:}|Z_k|\}$. Then, the likelihood that a given partition $\mathcal{U}_{\mathbf{X}}$ of a multi-object state $\mathbf{X}$, generates the detection set Z_k is

$$\sum_{\vartheta \in \Xi_k(\mathcal{L}(\mathcal{U}_{\mathbf{X}}))} \prod_{\mathbf{Y} \in \mathcal{U}_{\mathbf{X}}} \psi^{(\vartheta(\mathcal{L}(\mathbf{Y})))}_{Z_k}(\mathbf{Y}) \quad (7)$$

where

$$\psi^{(j)}_{\{z_{1:M}\}}(\mathbf{Y}) = \begin{cases} P_{D,k}(\mathbf{Y})\sigma_{D,k}(z_j|\mathbf{Y}), & \text{if } j = 1{:}M \\ 1 - P_{D,k}(\mathbf{Y}), & \text{if } j = 0 \end{cases},$$

with $\sigma_{D,k}(z_j|\mathbf{Y}) = g_{D,k}(z_j|\mathbf{Y})/\kappa_k(z_j)$ denoting the detection SNR for group $\mathbf{Y}$. The merged-measurement likelihood function is obtained by summing the group likelihoods (7) over all partitions of $\mathbf{X}$ [27]:

$$g_k(y_k|\mathbf{X}) \propto \sum_{\mathcal{U}_{\mathbf{X}} \in \mathcal{P}(\mathbf{X})} \sum_{\vartheta \in \Xi_k(\mathcal{L}(\mathcal{U}_{\mathbf{X}}))} \prod_{\mathbf{Y} \in \mathcal{U}_{\mathbf{X}}} \psi^{(\vartheta(\mathcal{L}(\mathbf{Y})))}_{D(y_k)}(\mathbf{Y}).$$

The multi-object filter (1)-(2) with merged-measurement likelihood is Bayes optimal in the sense that the filtering density contains all information on the current multi-object state in the presence of false positives, false negatives and occlusions. Unfortunately, this filter is numerically intractable due to the sum over all partitions of the multi-object state in the merged-measurement likelihood. At present, there is no polynomial time technique for truncating sums over partitions. Moreover, given a partition, computations involving the joint detection probability $P_{D,k}(\mathbf{Y})$, joint likelihood $g_{D,k}(z|\mathbf{Y})$ and associated joint densities are intractable except for scenarios with a few objects.

A GLMB approximation that reduces the number of partitions using the cluster structure of the predicted multi-object state and the sensor's resolution capabilities was proposed in [27]. Also, computation of joint densities are approximated by products of independent densities that minimise the Kullback-Leibler divergence [12]. Case studies on MOT with bearings only measurements shows very good tracking performance. Nonetheless, at present, this filter is still computationally demanding and therefore not suitable for online MOT with image data.

3. GLMB filter for tracking with image data

The GLMB filter (with the standard measurement likelihood) is a suitable candidate for online MOT [26, 28]. However, it is neither designed to handle occlusion nor image data. Even though occluded objects share the observations of the occluding objects, this situation is not permitted in the standard multi-object likelihood. Consequently, uncertainties in the states of occluded objects grow, while their existence probabilities quickly diminish to zero, leading to possible hi-jacking, and premature track termination in longer occlusions.

This section proposes an efficient *GLMB filter* that exploits information from image data and addresses false positives, false negatives and occlusions. Subsection 3.1 extends the standard observation model to allow occluded objects to share observations at the image level while Subsection 3.2 incorporates, into the death model, domain knowledge that mis-detected tracks with long durations are unlikely to disappear. The GLMB filter for image data and an efficient implementation are then described in Subsections 3.3 and 3.4.

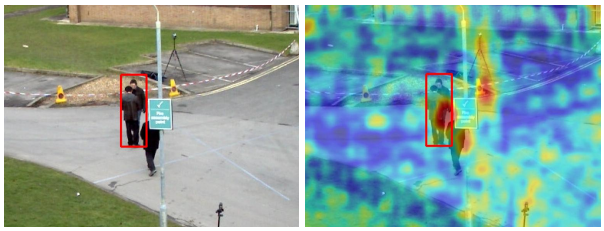

Figure 3: Example of shared measurement in mutual occlusion (left) and image observation for detection loss (right)

3.1. Hybrid Multi-Object Measurement Likelihood

While the detection set Z_k is an efficient compression of the image observation y_k, mis-detected (including occluded) objects will not be updated by the filter. On the other hand the uncompressed observation y_k contains relevant information about all objects, but updating with y_k is computationally expensive. Conceptually, we can have the best of both worlds by updating detected objects with the associated detections and mis-detected objects with the image observations localised to regions where these objects are expected. More importantly, this strategy exploits the fact that occluded objects share measurements with the objects occluding them.

To illustrate this we display the detection represented as a bounding box and image observation as a heatmap of some measure of confidence on the locations of the objects in Figure 3. The left of Figure 3 illustrates two types of false negatives due to 1) occlusion, and 2) detection loss. In this paper mutual occlusion is handled by shared measurements and detection loss is resolved by considering image observations as described in the right of Figure 3.

A hybrid tractable multi-object likelihood function that accommodates both detection and image observations can be obtained as follows. For tractability, it is assumed that each object generates observation independently from each other (similar to the standard observation model).

Given an object with state (x, ℓ) the likelihood of observing the local image $T(y_k)$ (some transformation of the image y_k) is $g_{T,k}(T(y_k)|x, \ell)$. On the other hand, given that there are no objects, the likelihood of observing $T(y_k)$ is $g_{T,k}(T(y_k)|\emptyset)$. The ratio

$$\sigma_{T,k}(T(y_k)|x, \ell) \triangleq \frac{g_{T,k}(T(y_k)|x, \ell)}{g_{T,k}(T(y_k)|\emptyset)} \tag{8}$$

can be interpreted as the image SNR (c.f. detection SNR Eq. (4)). For a given association map θ in the likelihood function Eq. (5), an object with state (x, ℓ) is mis-detected if $\theta(\ell) = 0$, in which case the value of $\psi_{Z_k}^{(\theta(\ell))}(x, \ell)$ is $1 - P_{D,k}(x, \ell)$, the probability of a miss. Consequently, after the Bayes update, track ℓ has no dependence on the observation y_k. In order for track ℓ to be updated with the local image $T(y_k)$, the value of $\psi_{D(y_k)}^{(\theta(\ell))}(x, \ell)$ should be scaled by the image SNR $\sigma_{T,k}(T(y_k)|x, \ell)$. Note that the value of $\psi_{D(y_k)}^{(\theta(\ell))}(x, \ell)$ should remain unchanged for $\theta(\ell) > 0$. Formally, this can be achieved by defining an extension of Eq. (6) as follows

$$\varphi_{y_k}^{(j)}(x, \ell) \triangleq \psi_{D(y_k)}^{(j)}(x, \ell) \left[\sigma_{T,k}(T(y_k)|x, \ell)\right]^{\delta_0[j]}. \tag{9}$$

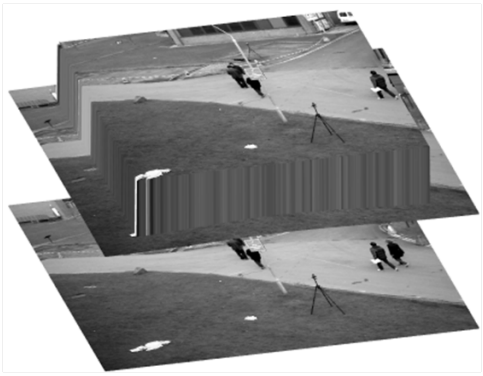

Figure 4: An example of a scene mask for the probability of survival. The border of scene has low probability of survival, but this probability increases as one moves further into the scene.

In other words, for $j = 0$, $\varphi_{y_k}^{(j)}(x, \ell)$ is equal to the image SNR Eq. (8) scaled by the probability that its not detected, otherwise it is equal to the detection SNR Eq. (4) scaled by the detection probability.

Given a state (x, ℓ), $\varphi_{y_k}^{(\theta(\ell))}(x, \ell)$ plays the same role as $\psi_{Z_k}^{(\theta(\ell))}(x, \ell)$, but accommodates both detection measurements and image measurements. Moreover, since each object generates observation independently from each other, the hybrid multi-object likelihood function has the same form as Eq. (5), but with $\psi_{D(y_k)}^{(\theta(\ell))}(x, \ell)$ replaced by $\varphi_{y_k}^{(\theta(\ell))}(x, \ell)$, i.e.

$$g_k(y_k|\mathbf{X}_k) \propto \sum_{\theta \in \Theta_k(\mathcal{L}(\mathbf{X}_k))} \prod_{(x,\ell) \in \mathbf{X}_k} \varphi_{y_k}^{(\theta(\ell))}(x, \ell). \tag{10}$$

In visual occlusions, the hybrid likelihood allows occluded objects to share the image observations of the objects that occlude them. Moreover, when integrated into the Bayes recursion Eq. (1)-(2), consideration for a track-length-dependent survival probability in combination with information from the image observation, reduces uncertainties in the states of occluded objects and maintains their existence probabilities to keep the tracks alive. Hence, hi-jacking and premature track termination in longer occlusions will be avoided.

Remark: The hybrid multi-object likelihood function Eq. (10) is a generalization of both the standard multi-object likelihood and the separable likelihood in [10]. When $P_{D,k}(x, \ell) = 1$ for each $(x, \ell) \in \mathbf{X}_k$, i.e. there is no false negative, the hybrid likelihood Eq. (10) is the same as the standard likelihood Eq. (5). On the other hand, if $P_{D,k}(x, \ell) = 0$ for each $(x, \ell) \in \mathbf{X}_k$, i.e. there is no detection, then the only non-zero term in the hybrid likelihood (10) is one with $\theta(\ell) = 0$ for all $\ell \in \mathcal{L}(\mathbf{X}_k)$. In this case, the hybrid likelihood Eq. (10) reduces to the separable likelihood in [10]. For a general detection profile $P_{D,k}$, the hybrid likelihood Eq. (10) reduces to the standard likelihood Eq. (5) when $\sigma_{T,k}(T(y_k)|x, \ell) = 1$ for each $(x, \ell) \in \mathbf{X}_k$.

Note that a hybrid likelihood function can be also developed for the merged-measurement model. However, the resulting multi-object filter still suffers from the same intractability as the merged-measurement filter.

3.2. Death model

In most video MOT applications, if an object stays in the scene for a long time, then it is more likely to continue to do so, provided it is not close to the designated exit regions. Such prior empirical knowledge can be used to improve occlusion handling, especially long occlusions that can lead to premature track termination in on-line MOT algorithms. In general, the GLMB filter would terminate an object that has not been detected over several frames. However, if this object has been in the scene for some time and is not in the proximity of designated exit regions, then it is highly likely to be occluded and track termination should be delayed. The labeled RFS formulation enables such prior information to be incorporated into track termination in a principled manner, via the survival probability.

The labeled RFS formulation accommodates survival probabilities that depend on track lengths since a labeled state contains the time of birth in its label, and the track length is simply the difference between the current time and the time of birth. In practice, it is unlikely for an object to disappear in the middle of the visual scene (even if detection loss or occluded) whereas it is more likely to disappear near designated exit regions due to the scene structure (e.g. the borders of the scene). Hence, we require the survival probability to be large (close to one) in the middle of the scene and small (close to zero) on the edges or designated death regions. The scene structure is reflected into the state dependent survival probability by introducing a scene mask that shapes the survival probability over the surveillance region. Furthermore, since objects staying in the scene for a long time are more certain to continue existing, we require the survival probability to increase to one as its track length increases.

An explicit form of the survival probability that satisfies these requirements is given by

$$P_{S,k}(x,\ell) = \frac{b(x)}{1+\exp(-\gamma(k-\ell[1,0]^{\mathrm{T}}))} \tag{11}$$

where $b(x)$ is a scene mask that represents the scene structure, e.g., entrance or exit as illustrated in Figure 4, γ is a control parameter of the sigmoid function that reflects the expected length of the object trajectory. The scene mask $b(x)$ can be learned from a set of training data or designed from the known scene structure.

3.3. GLMB Recursion

A GLMB density can be written in the following form

$$\pi(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{\xi\in\Xi} \sum_{I\subseteq\mathbb{L}} \omega^{(I,\xi)} \delta_I[\mathcal{L}(\mathbf{X})] \left[p^{(\xi)}\right]^{\mathbf{X}}, \tag{12}$$

where each $\xi \in \Xi$ represents a history of association maps $\xi = (\theta_{1:k})$, each $p^{(\xi)}(\cdot,\ell)$ is a probability density on $\mathbb{X}$, and each $\omega^{(I,\xi)}$ is non-negative with $\sum_{\xi\in\Xi}\sum_{I\subseteq\mathbb{L}} \omega^{(I,\xi)} = 1$. The cardinality distribution of a GLMB is given by

$$\Pr(|\mathbf{X}| = n) = \sum_{\xi\in\Xi} \sum_{I\subseteq\mathbb{L}} \delta_n\left[|I|\right] \omega^{(I,\xi)}, \tag{13}$$

while, the existence probability and probability density of track $\ell \in \mathbb{L}$ are respectively

$$r(\ell) = \sum_{\xi\in\Xi} \sum_{I\subseteq\mathbb{L}} 1_I(\ell)\omega^{(I,\xi)}, \tag{14}$$

$$p(x,\ell) = \frac{1}{r(\ell)} \sum_{\xi\in\Xi} \sum_{I\subseteq\mathbb{L}} 1_I(\ell)\omega^{(I,\xi)} p^{(\xi)}(x,\ell). \tag{15}$$

Given the GLMB density Eq. (12), an intuitive multi-object estimator is the *multi-Bernoulli estimator*, which first determines the set of labels $L \subseteq \mathbb{L}$ with existence probabilities above a prescribed threshold, and second the MAP/mean estimates from the densities $p(\cdot,\ell), \ell \in L$, for the states of the objects. A popular estimator is a suboptimal version of the Marginal Multi-object Estimator [18], which first determines the pair (L,ξ) with the highest weight $\omega^{(L,\xi)}$ such that $|L|$ coincides with the MAP cardinality estimate, and second the MAP/mean estimates from $p^{(\xi)}(\cdot,\ell), \ell \in L$, for the states of the objects.

The GLMB family enjoys a number of nice analytical properties. The void probability functional–a necessary and sufficient statistic–of a GLMB, the Cauchy-Schwarz divergence between two GLMBs, the L_1-distance between a GLMB and its truncation, can all be computed in closed form [26]. The GLMB is flexible enough to approximate any labeled RFS density with matching intensity function and cardinality distribution [12]. More importantly, the GLMB family is closed under the prediction equation (Eq. 1) and conjugate with respect to the standard observation likelihood [8].

In the following we show that the GLMB family is conjugate with respect to the hybrid observation likelihood function. Hence, starting from an initial GLMB prior, all multi-object predicted and updated densities propagated by the Bayes recursion Eq. (1)-(2) are GLMBs. For notational compactness, we drop the subscript k for the current time, the next time is indicated by the subscript '+'.

Proposition 1. *Suppose that the multi-object prediction density to time $k+1$ is a GLMB of the form*

$$\bar{\pi}_+(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{\xi,I_+} \bar{\omega}_+^{(\xi,I_+)} \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[\bar{p}_+^{(\xi)}\right]^{\mathbf{X}_+}, \tag{16}$$

where $\xi \in \Xi$, $I_+ \in \mathcal{F}(\mathbb{L}_+)$. Then under the hybrid observation likelihood function Eq. (10), the filtering density at time $k+1$ is a GLMB of the form

$$\pi_{y_+}(\mathbf{X}_+) \propto \Delta(\mathbf{X}_+) \sum_{\xi,I_+,\theta_+} \omega_{y_+}^{(\xi,I_+,\theta_+)} \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[p_+^{(\xi,\theta_+)}\right]^{\mathbf{X}_+}, \tag{17}$$

where $\theta_+ \in \Theta_+$, and

$$\omega_{y_+}^{(\xi,I_+,\theta_+)} = \bar{\omega}_+^{(\xi,I_+)} 1_{\Theta_+(I_+)}(\theta_+) \left[\bar{\varphi}_{y_+}^{(\xi,\theta_+)}\right]^{I_+}, \tag{18}$$

$$\bar{\varphi}_{y_+}^{(\xi,\theta_+)}(\ell_+) = \left\langle \bar{p}_+^{(\xi)}(\cdot,\ell_+), \varphi_{y_+}^{(\theta_+(\ell_+))}(\cdot,\ell_+) \right\rangle, \tag{19}$$

$$p_+^{(\xi,\theta_+)}(x_+,\ell_+) = \frac{\bar{p}_+^{(\xi)}(x_+,\ell_+)\varphi_{y_+}^{(\theta_+(\ell_+))}(x_+,\ell_+)}{\bar{\varphi}_{y_+}^{(\xi,\theta_+)}(\ell_+)}. \tag{20}$$

Proof. Note that the likelihood function Eq. (10) at time $k+1$ can be written as

$$g_+(y_+|\mathbf{X}_+) \propto \sum_{\theta_+} 1_{\Theta_+(\mathcal{L}(\mathbf{X}_+))}(\theta_+) \left[\tilde{\varphi}_{y_+}^{(\theta_+)}\right]^{\mathbf{X}_+},$$

where $\tilde{\varphi}_{y_+}^{(\theta_+)}(x,\ell) \triangleq \varphi_{y_+}^{(\theta_+(\ell))}(x,\ell)$.
Using Bayes rule

$$\begin{aligned}
\pi_{y_+}(\mathbf{X}_+) &= \bar{\pi}_+(\mathbf{X}_+) g_+(y_+|\mathbf{X}_+) \\
&\propto \Delta(\mathbf{X}_+) \sum_{\xi,I_+} \bar{\omega}_+^{(\xi,I_+)} \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[\bar{p}_+^{(\xi)}\right]^{\mathbf{X}_+} \\
&\quad \times \sum_{\theta_+} 1_{\Theta_+(\mathcal{L}(\mathbf{X}_+))}(\theta_+) \left[\tilde{\varphi}_{y_+}^{(\theta_+)}\right]^{\mathbf{X}_+} \\
&= \Delta(\mathbf{X}_+) \sum_{\xi,I_+,\theta_+} \bar{\omega}_+^{(\xi,I_+)} 1_{\Theta_+(I_+)}(\theta_+) \\
&\quad \times \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[\bar{p}_+^{(\xi)} \tilde{\varphi}_{y_+}^{(\theta_+)}\right]^{\mathbf{X}_+} \\
&= \Delta(\mathbf{X}_+) \sum_{\xi,I_+,\theta_+} \bar{\omega}_+^{(\xi,I_+)} 1_{\Theta_+(I_+)}(\theta_+) \\
&\quad \times \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[\bar{\varphi}_{y_+}^{(\xi,\theta_+)}\right]^{\mathcal{L}(\mathbf{X}_+)} \left[\frac{\bar{p}_+^{(\xi)} \tilde{\varphi}_{y_+}^{(\theta_+)}}{\bar{\varphi}_{y_+}^{(\xi,\theta_+)}}\right]^{\mathbf{X}_+} \\
&= \Delta(\mathbf{X}_+) \sum_{\xi,I_+,\theta_+} \bar{\omega}_+^{(\xi,I_+)} 1_{\Theta_+(I_+)}(\theta_+) \left[\bar{\varphi}_{y_+}^{(\xi,\theta_+)}\right]^{I_+} \\
&\quad \times \delta_{I_+}[\mathcal{L}(\mathbf{X}_+)] \left[p_+^{(\xi,\theta_+)}\right]^{\mathbf{X}_+}.
\end{aligned}$$

□

In this work we adopt the joint prediction and update strategy [28] for the proposed video MOT GLMB filter. Using the same line of arguments as in [28], yields the following recursion

Proposition 2. *Given the GLMB filtering density (12) at time k, the filtering density at time $k+1$ is:*

$$\pi_+(\mathbf{X}) \propto \Delta(\mathbf{X}) \sum_{I,\xi,I_+,\theta_+} \omega^{(I,\xi)} \omega_{y_+}^{(I,\xi,I_+,\theta_+)} \delta_{I_+}[\mathcal{L}(\mathbf{X})] \left[p_{y_+}^{(\xi,\theta_+)}\right]^{\mathbf{X}}, \tag{21}$$

where $I \in \mathcal{F}(\mathbb{L})$, $\xi \in \Xi$, $I_+ \in \mathcal{F}(\mathbb{L}_+)$, $\theta_+ \in \Theta_+(I_+)$, and

$$\begin{aligned}
\omega_{y_+}^{(I,\xi,I_+,\theta_+)} &= \left[1 - \bar{P}_S^{(\xi)}\right]^{I-I_+} \left[\bar{P}_S^{(\xi)}\right]^{I\cap I_+} \\
&\quad \times \left[1 - r_{B,+}\right]^{\mathbb{B}_+ - I_+} r_{B,+}^{\mathbb{B}_+ \cap I_+} \left[\bar{\varphi}_{y_+}^{(\xi,\theta_+)}\right]^{I_+}, && (22) \\
\bar{P}_S^{(\xi)}(\ell) &= \left\langle p^{(\xi)}(\cdot,\ell), P_S(\cdot,\ell) \right\rangle, && (23) \\
\bar{\varphi}_{y_+}^{(\xi,\theta_+)}(\ell_+) &= \left\langle \bar{p}_+^{(\xi)}(\cdot,\ell_+), \varphi_{y_+}^{(\theta_+(\ell_+))}(\cdot,\ell_+) \right\rangle, && (24) \\
\bar{p}_+^{(\xi)}(x_+,\ell_+) &= 1_{\mathbb{L}}(\ell_+) \frac{\left\langle P_S(\cdot,\ell_+) f_+(x_+|\cdot,\ell_+), p^{(\xi)}(\cdot,\ell_+) \right\rangle}{\bar{P}_S^{(\xi)}(\ell_+)} \\
&\quad + 1_{\mathbb{B}_+}(\ell_+) p_{B,+}(x_+,\ell_+), && (25) \\
p_+^{(\xi,\theta_+)}(x_+,\ell_+) &= \frac{\bar{p}_+^{(\xi)}(x_+,\ell_+) \varphi_{y_+}^{(\theta_+(\ell_+))}(x_+,\ell_+)}{\bar{\varphi}_{y_+}^{(\xi,\theta_+)}(\ell_+)}. && (26)
\end{aligned}$$

The summation in Eq. (21) can be interpreted as an enumeration of all possible combinations of births, deaths and survivals together with associations of new measurements to hypothesized tracks. Observe that Eq. (21) does indeed take on the same form as Eq. (12) when rewritten as a sum over I_+, ξ, θ_+ with weights

$$\omega_+^{(I_+,\xi,\theta_+)} \propto \sum_I \omega^{(I,\xi)} \omega_{y_+}^{(I,\xi,I_+,\theta_+)}. \tag{27}$$

Hence at the next iteration we only propagate forward the components (I_+, ξ, θ_+) with weights $\omega_+^{(I_+,\xi,\theta_+)}$.

Remark: It is also possible to approximate the resulting GLMB filtering density by an LMB with matching 1st moment and cardinality distribution [29]. This so-called LMB filtering strategy reduces considerable computations since an LMB is a GLMB with one term. However, tracking performance tend to degrade, especially in scenarios with many closely space objects. Note that for high SNR scenarios the detection probability is high, hence the recursion Eq. (21)-(27) would process detections mostly. On the other hand when the detection probability is low it would process the image mostly. In practice the SNR varies between different regions in the observation space as well as with time, the recursion Eq. (21)-(27) adaptively processes detections and image data to improve performance while reducing the computational cost.

3.4. GLMB Filter Implementation

The number of terms in the GLMB filtering density grows super-exponentially, and it is necessary to truncate these terms without exhaustive enumeration. A two-stage implementation of the GLMB filter truncates the prediction and filtering densities using the K-shortest path and the ranked assignment algorithms, respectively [26]. In [28] the joint prediction and update was designed to improve the truncation efficiency of the two-staged implementation. Further, the GLMB truncation can be performed via Gibbs sampling with linear complexity in the number of detections (the reader is referred to [28] for derivations and analysis). Fortuitously, this implementation can be readily adapted for the video MOT GLMB filter Eq. (21)-(27).

The GLMB filtering density Eq. (12) at time k is completely characterized by the parameters $(\omega^{(I,\xi)}, p^{(\xi)})$, $(I,\xi) \in \mathcal{F}(\mathbb{L})\times\Xi$, which can be enumerated as $\{(I^{(h)}, \xi^{(h)}, \omega^{(h)}, p^{(h)})\}_{h=1}^H$, where

$$\omega^{(h)} \triangleq \omega^{(I^{(h)},\xi^{(h)})}, \; p^{(h)} \triangleq p^{(\xi^{(h)})}.$$

Since Eq. (12) can now be rewritten as

$$\pi(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{h=1}^{H} \omega^{(h)} \delta_{I^{(h)}}[\mathcal{L}(\mathbf{X})] \left[p^{(h)}\right]^{\mathbf{X}},$$

implementing the GLMB filter amounts to propagating the component set $\{(I^{(h)}, \omega^{(h)}, p^{(h)})\}_{h=1}^H$ (there is no need to store $\xi^{(h)}$) forward in time using Eq. (21)-(27). The procedure for computing the component set $\{(I_+^{(h_+)}, \omega_+^{(h_+)}, p_+^{(h_+)})\}_{h_+=1}^{H_+}$ at the next time is summarized in Algorithm 1. Note that to be consistent with the

indexing by h instead of (I,ξ), we also abbreviate

$$\begin{aligned}
\bar{P}_S^{(h)}(\ell_i) &\triangleq \bar{P}_S^{(\xi^{(h)})}(\ell_i), \quad \bar{p}_+^{(h)}(x,\ell_i) \triangleq \bar{p}_+^{(\xi^{(h)})}(x,\ell_i), \\
\bar{\varphi}_{y_+}^{(h,j)}(\ell_i) &\triangleq \left\langle \bar{p}_+^{(h)}(\cdot,\ell_i), \varphi_{y_+}^{(j)}(\cdot,\ell_i) \right\rangle, \\
\eta_i^{(h)}(j) &\triangleq \begin{cases} 1-\bar{P}_S^{(h)}(\ell_i), & \ell_i \in I^{(h)},\ j<0, \\ \bar{P}_S^{(h)}(\ell_i)\bar{\varphi}_{y_+}^{(h,j)}(\ell_i), & \ell_i \in I^{(h)},\ j\geq 0, \\ 1-r_{B+}(\ell_i), & \ell_i \in \mathbb{B}_+,\ j<0, \\ r_{B+}(\ell_i)\bar{\varphi}_{y_+}^{(h,j)}(\ell_i), & \ell_i \in \mathbb{B}_+,\ j\geq 0. \end{cases}
\end{aligned} \tag{28}$$

Algorithm 1. Joint Prediction and Update

- input: $\{(I^{(h)}, \omega^{(h)}, p^{(h)})\}_{h=1}^{H}$, y_+, Z_+, $H_+^{\max}$,
- input: $\{(r_{B,+}^{(\ell)}, p_{B,+}^{(\ell)})\}_{\ell\in\mathbb{B}_+}$, P_S, f_+, $\sigma_{D,+}$, $\sigma_{T,+}$,
- output: $\{(I_+^{(h_+)}, \omega_+^{(h_+)}, p_+^{(h_+)})\}_{h_+=1}^{H_+}$

sample counts $[T_+^{(h)}]_{h=1}^{H}$ from multinomial distribution with parameters $H_+^{\max}$ trials and weights $[\omega^{(h)}]_{h=1}^{H}$
for $h = 1:H$
  compute $\eta^{(h)} := [\eta_i^{(h)}(j)]_{(i,j)=(1,-1)}^{(|I^{(h)}\cup\mathbb{B}_+|,|Z_+|)}$ using Eq. (28)
  initialize $\gamma^{(h,1)}$
  $\{\gamma^{(h,t)}\}_{t=1}^{\tilde{T}_+^{(h)}} :=$ Unique(Gibbs($\gamma^{(h,1)}, T_+^{(h)}, \eta^{(h)}$));
  for $t = 1:\tilde{T}_+^{(h)}$
    compute $I_+^{(h,t)}$ from $I^{(h)}$ and $\gamma^{(h,t)}$ using Eq. (29)
    compute $\omega_+^{(h,t)}$ from $\omega^{(h)}$ and $\gamma^{(h,t)}$ using Eq. (30)
    compute $p_+^{(h,t)}$ from $p^{(h)}$ and $\gamma^{(h,t)}$ using Eq. (31)
  end
end
$(\{(I_+^{(h_+)}, p_+^{(h_+)})\}_{h_+=1}^{H_+}, \sim, [U_{h,t}])$
  $:=$ Unique($\{(I_+^{(h,t)}, p_+^{(h,t)})\}_{(h,t)=(1,1)}^{(H,\tilde{T}_+^{(h)})}$);
for $h_+ = 1:H_+$
  $\omega_+^{(h_+)} := \sum_{h,t:U_{h,t}=h_+} \omega_+^{(h,t)}$;
end
normalize weights $\{\omega_+^{(h_+)}\}_{h_+=1}^{H_+}$

Algorithm 1a. Gibbs

- input: $\gamma^{(1)}, T, \eta = [\eta_i(j)]$
- output: $\gamma^{(1)}, ..., \gamma^{(T)}$

$P :=$ size(η, 1); $M :=$ size(η, 2) $- 2$; $c := [-1:M]$;
for $t = 2:T$
  $\gamma^{(t)} := [\,]$;
  for $n = 1:P$
    for $j = 1:M$
      $\eta_n(j) := \eta_n(j)(1 - 1_{\{\gamma_{1:n-1}^{(t)}, \gamma_{n+1:P}^{(t-1)}\}}(j))$;
    end
    $\gamma_n^{(t)} \sim$ Categorical(c, η_n); $\gamma^{(t)} := [\gamma^{(t)}, \gamma_n^{(t)}]$;
  end
end

There are three main steps in one iteration of the GLMB filter.

Step 1. First, the Gibbs sampler (Algorithm 1a) is used to generate the *auxiliary vectors* $\gamma^{(h,t)}$, $h = 1:H$, $t = 1:\tilde{T}_+^{(h)}$, with the most significant weights $\omega_+^{(h,t)}$(note that $\gamma^{(h,t)}$ is an equivalent representation of the hypothesis $(I_+^{(h,t)}, \theta_+^{(h,t)})$, with components $\gamma_i^{(h,t)}$, $i = 1:|I^{(h)}\cup\mathbb{B}_+|$, defined as $\theta_+^{(h,t)}(\ell_i)$ when $\ell_i \in I^{(h)}$, and -1 otherwise [28]). The Gibbs sampler has an exponential convergence rate [28]. More importantly, it is not necessary to discard burn-ins and wait for samples from the stationary distribution. All distinct samples can be used, the larger the weights, the smaller the L_1 error from the true GLMB filtering density [28].
Step 2. Second, the auxiliary vectors are used to generate an intermediate set of parameters with the most significant weights $(I^{(h)}, I_+^{(h,t)}, \omega_+^{(h,t)}, p_+^{(h,t)})$, $h = 1:H$, $t = 1:\tilde{T}_+^{(h)}$, via Eq. (21). Note that given a component h and $\gamma^{(h,t)}$, it can be shown that [28]

$$I_+^{(h,t)} = \{\ell_i \in I^{(h)} \cup \mathbb{B}_+ : \gamma_i^{(h,t)} \geq 0\}, \tag{29}$$

$$\omega_+^{(h,t)} \propto \omega^{(h)} \prod_{i=1}^{|I^{(h)}\cup\mathbb{B}_+|} \eta_i^{(h)}(\gamma_i^{(h,t)}), \tag{30}$$

$$p_+^{(h,t)}(\cdot,\ell_i) = \frac{\bar{p}_+^{(h)}(\cdot,\ell_i)\varphi_{y_+}^{(\gamma_i^{(h,t)})}(\cdot,\ell_i)}{\bar{\varphi}_{y_+}^{(h,\gamma_i^{(h,t)})}(\ell_i)}. \tag{31}$$

Note also that $\theta_+^{(h,t)}(\ell_i) = \gamma_i^{(h,t)}$ when $\gamma_i^{(h,t)} \geq 0$, for $\ell_i \in I_+^{(h,t)}$.
Step 3. Third, the intermediate parameters are marginalized via Eq. (27) to give the new parameter set $\{(I_+^{(h_+)}, \omega_+^{(h_+)}, p_+^{(h_+)})\}_{h_+=1}^{H_+}$. Note that $U_{h,t}$ gives the index of the GLMB component at time $k+1$ that $(I^{(h)}, I_+^{(h,t)}, p_+^{(h,t)})$ contributes to.

4. Experimental results

The proposed MOT filter is tested on a simulated TBD application in subsection 4.1, and on real video data in subsection 4.2.

4.1. TBD

4.1.1. Dynamic motion and observation model

Consider a scenario with upto 5 objects, each with a 4D state $x_k = [\ p_{x,k}, \dot{p}_{x,k}, p_{y,k}, \dot{p}_{y,k}\ ]^{\mathrm{T}}$ of position and velocity. Each object follows a constant velocity model with Gaussian transition density

$$f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x_k; Fx_{k-1}, Q),$$

where

$$F = I_2 \otimes \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix},$$

I_2 is the 2×2 identity matrix, $\otimes$ denotes the Kronecker product, T_s is the sampling period of the video data, $Q = \sigma_v^2 I_2$, and $\sigma_v = 1$ pixels/frame is the noise standard deviation.

The birth density is assumed to be LMB with 5 components of 0.03 birth probability and Gausssian distributed birth densities as

$$\begin{aligned}
&\mathcal{N}(\cdot,[5;0;5;0]^T,P_\gamma), \quad \mathcal{N}(\cdot,[5;0;25;0]^T,P_\gamma), \\
&\mathcal{N}(\cdot,[5;0;90;0]^T,P_\gamma), \quad \mathcal{N}(\cdot,[90;0;30;0]^T,P_\gamma), \\
&\mathcal{N}(\cdot,[80;0;90;0]^T,P_\gamma), \quad P_\gamma = diag([3;2;3;2]).
\end{aligned}$$

The survival probability P_S for the standard GLMB filter is 0.98 and the control parameter γ of the age-dependent survival probability is set to 0.1. The scene mask $b(x)$ of the same shape as Figure 4 with a margin of 10 pixels around the border area is used.

The observations are raw images simulated from the radar TBD measurement model [11], consisting of an array of pixel values representing the power signal returns i.e., $y_k = [y^{(1)}, ..., y^{(i)}]$, with

$$y^{(i)} = \left| \sum_{\mathbf{x}\in\mathbf{X}: i\in C(\mathbf{x})} A(\mathbf{x}) h_A^{(i)}(\mathbf{x}) + w^{(i)} \right|^2, \tag{32}$$

where $C(\mathbf{x})$ is usually referred to as the target template, $A(\mathbf{x})$ denotes the amplitude of the return signal. Setting a relatively high SNR for the simulation means that the filter will mostly operate like a standard GLMB filter, while a low SNR means it mostly operates like a TBD-GLMB filter. Neither scenarios are interesting. In this example we simulate the observations with SNRs that fluctuate between 10dB and 7dB within the same image. Further, to demonstrate how the tracker adapts to the SNR mismatch, the observation model used by the tracker is instantiated with a 10dB SNR. The point spread function in cell i from state $\mathbf{x}$ is given by

$$h_A^{(i)}(\mathbf{x}) = \exp\left(-\frac{(r_i - r(\mathbf{x}))^2}{2R} - \frac{(s_i - s(\mathbf{x}))^2}{2S}\right), \tag{33}$$

where $R = 1$ and $S = 1$ are constants related to the image cell resolution; $r(\mathbf{x})$ and $s(\mathbf{x})$ are the coordinates of the object in the measurement space; r_i and s_i are the cell centroids. The detection and transformed image observation for the raw pixel image model Eq. (32)-(33) are obtained as follows.

A hard thresholding is applied to the raw image y_k, and the detection model used by the proposed filter consists of a single-object detection likelihood $g_{D,k}(z|x,\ell) = \mathcal{N}(z; Hx, \Sigma)$, where $H = [1\ 0\ 0\ 0;\ 0\ 0\ 1\ 0]$; $\Sigma = diag(4^2, 4^2)$, a detection probability P_D of 0.98, and a clutter rate of 10 points per frame. On the other hand, the transformed image $T(y_k)$ is the correlation response between the reference template and the observed template, obtained from the raw image y_k via Kernelized Correlation Filtering (KCF) [30]. The image observation model use by the proposed filter is given by

$$g_{T,k}(T(y_k)|x,\ell) \propto \exp\left(-\frac{1}{\sigma^2}\left(\left\|\mathrm{f}(x) - \bar{\mathrm{f}}_\ell\right\|^2\right)\right),$$

where $\mathrm{f}(x)$ is the observed template at a given object state x; $\bar{\mathrm{f}}_\ell$ is the reference template of the track label ℓ (consists of pixel intensities in a 3 pixel by 3 pixel region); σ controls the shape of the function. The information flow for the proposed hybrid observation likelihood with KCF is illustrated in Figure 5. The Unscented Transform is used for the measurement update for image observations. To adapt appearance changes, pixels inside regions with confident point detections are used to update $\bar{\mathrm{f}}_\ell$. The reference template is not updated when no detection is assigned. In the case of mutual occlusion, the reference template is also not updated, however, a detection from the occluder

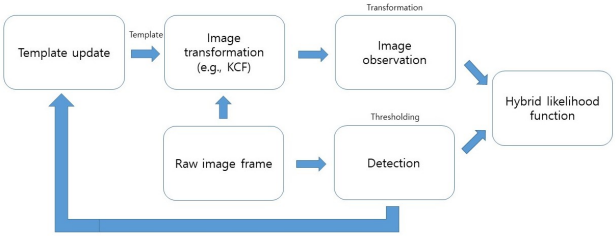

Figure 5: A flow diagram for hybrid likelihood function

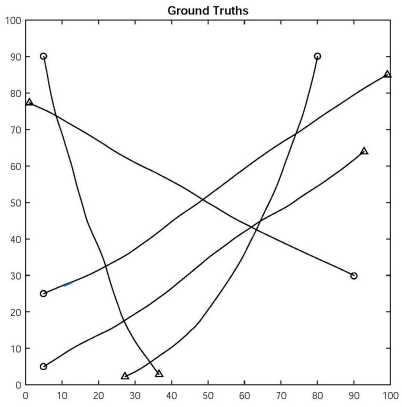

Figure 6: True tracks in the x y plane. Start/Stop positions are shown with ∘/△.

(other target) is used as a shared measurement (see Section 3.1) for the occluded track update until occlusion ends. The track is indicated as occluded when there exists a detection with different appearance (i.e., different reference template) whose overlapped target extent is over the pre-defined threshold (0.4 in our experiments). Empirically, this strategy is more robust than the update scheme in [30] because accumulated learning errors are reduced by updating the model with confident detections. Note that further improvements in observation modelling and template update can be adopted such as [31], however, it requires non-negligible computations.

4.1.2. Simulation scenario and comparison results

The size of the surveillance area is 100 pixels by 100 pixels and the size of the image cell is 1. Image data for the true tracks (shown in Figure 6) is generated according the observation model Eq. (32)-(33). Sample snap shots of image sequence are displayed in Figure 7 together with the true number of objects and the description of detection results (by hard-thresholding) for each snapshot. Figure 7 illustrates that low SNR images are prone to false negatives, and that merged detections occur in mutual occlusions.

We compare the standard GLMB filter (GLMB) with the proposed GLMB for image observation (GLMB-IM) (with time-dependent survival probability). The performance comparison is summarized in Figure 8 with respect to OSPA errors [32] calculated over 100 Monte Carlo runs.

Note from Figure 8, that the standard GLMB filter quickly

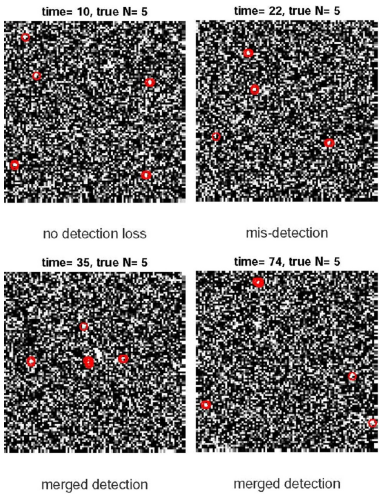

Figure 7: 4 snaps shots from the image data sequence superimposed with detections (red circles), which show no detection loss, detection loss and merged detections.

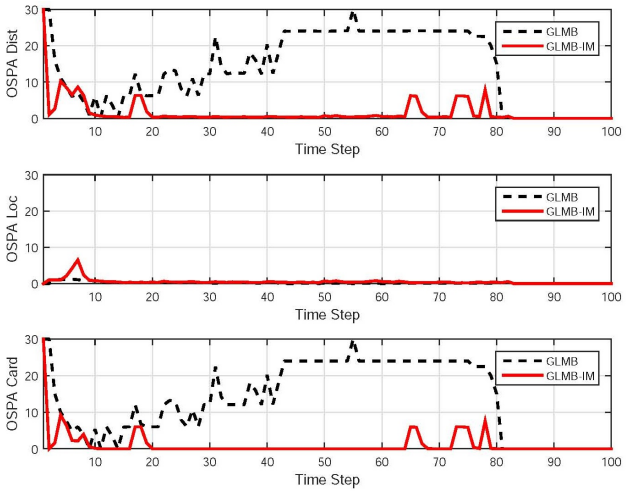

Figure 8: OSPA Error for two filters: GLMB vs GLMB-IM (proposed) (first row: overall OSPA distance, Second row: localization error, third row: cardinality error)

lost tracks due to the false negatives from low SNR or merged detections from object occlusions. On the other hand, the GLMB-IM filter keeps the tracks due to the combination of proposed survival probability and effective measurement updates from the image data.

4.2. Visual Tracking

4.2.1. Dataset and parameter settings

In this subsection, we test the proposed MOT filter on the *PETS 2009* dataset [33]; the *MOTChallenge* dataset [34]. To benchmark the tracking performance against a number of recent algorithms, we use published detection results and evaluation tools from [34]. The motion model is learned from the training dataset by considering the maximum speed of the object with regard to the frame rate.

Remark: While the object's extent such as its bounding box [1], [5], can be included in the object state, effective modeling of extent dynamics is application dependent. In experiments we estimate an object's extent via the median values of the x, y scale of the detections associated with existing tracks in a given time window.

Remark: Similar to single-object visual tracking filtering in [1], the predicted covariance for each track is capped to a prescribed value to prevent it from exploding over time.

The RFS framework accommodates a time-varying birth model. In this experiment, we use a birth model that consists of both static and dynamic components. The static component is an LMB that describes expected locations where objects are highly likely to appear e.g., the image border/footpaths near the image border. The dynamic component is a time-varying LMB that exploits measurements with weak associations (to existing tracks) to describe highly likely object births at the next time frame [29].

The detection $z \in D(y_k)$ of an object is obtained by a detector based on discriminative part-baed model (DPM) [35] and the same point measurement model in the numerical example is used with $\Sigma = diag(5^2, 5^2)$. The probability of detection P_D is 0.98 and the clutter rate is 5, i.e., an average of 5 clutter measurements per frame. These parameters can be obtained from training data or learned on-the-fly in the RFS framework as proposed in [36]. For image observations, similar to the TBD example in Section 4.1.1,we also applied KCF but to the Histogram of Oriented Gradients (HOG) feature image instead of the raw pixel image [30]. The template update strategy described in Section 4.1.1 is used. More advanced template learning and update algorithm can be adapted to handle re-identification of people [37], however, it is the beyond scope of this paper.

In the experiments, the maximum number of track hypotheses $H_+^{\max}$ is set to 200 for a good balance between the accuracy and computational efficiency, and track estimates are obtained from the GLMB filtering density via the LMB estimator described in Subsection 3.3. Note that when the LMB estimator terminates a track, the GLMB filtering density still contains its existence probability and state density (hence state estimate). This information is completely deleted only when its existence probability is so negligible that all relevant GLMB components are truncated. If not completely deleted, it is possible that due to new evidence in the data at later time, a track's existence probability becomes significant enough to be selected by multi-object estimator, leading to track fragmentation. While this problem can be addressed in a principled manner via multi-object smoothing, the GM-PHD smoother [38] is not applicable and an implementation of the forward-backward GLMB smoother [39] is not yet available. Nonetheless, we can exploit the sequence of measurements associated with each track in the GLMB framework, to smooth the estimated tracks and recover missing state estimates.

Table 1: Tracking performance on the PETS 2009 dataset (Online methods are indicated by $*$)

Sequence	Tracker	MOTA ↑	MOTP ↑	MT ↑	ML ↓	FM ↓	IDS ↓
S2L1	GLMB-IM (Ours) $*$	91.0 %	77.2 %	**18**	0	20	15
	GLMB [8] $*$	83.9 %	76.1 %	16	0	70	50
	MHT_AM [40]	**92.6%**	**79.1%**	18	0	**12**	**13**
	MHT [41]	84.1%	77.5%	17	0	65	45
	CEM [5]	90.3 %	74.3 %	18	0	15	22
S2L2	GLMB-IM (Ours) $*$	55.4 %	57.8 %	10	2	163	133
	GLMB [8] $*$	33.4 %	56.3 %	4	6	275	162
	MHT_AM [40]	**59.2%**	**61.4%**	10	2	162	**120**
	MHT [41]	38.0 %	58.8 %	3	8	273	154
	CEM [5]	58.1 %	59.8 %	**11**	**1**	**153**	167
S2L3	GLMB-IM (Ours) $*$	37.5 %	69.4 %	**12**	**19**	22	30
	GLMB [8] $*$	35.5 %	65.6 %	8	19	23	32
	MHT_AM [40]	38.5 %	**70.8%**	9	22	**9**	**8**
	MHT [41]	**40.8** %	67.3 %	10	21	19	18
	CEM [5]	39.8 %	65.0 %	8	19	22	27
S1L1-2	GLMB-IM (Ours) $*$	61.2 %	69.0 %	21	15	21	13
	GLMB [8] $*$	55.1 %	63.0 %	19	14	60	20
	MHT_AM [40]	**62.1%**	**70.3%**	21	9	14	**11**
	MHT [41]	61.6 %	68.0 %	**22**	12	23	31
	CEM [5]	52.0 %	66.5 %	17	14	52	41
S1L2-1	GLMB-IM (Ours) $*$	25.4 %	59.5 %	3	25	34	26
	GLMB [8] $*$	23.8 %	56.0 %	3	25	35	26
	MHT_AM [40]	25.4 %	**62.2%**	3	24	30	**25**
	MHT [41]	24.0 %	58.4 %	**5**	23	**29**	33
	CEM [5]	**29.6%**	58.8 %	2	**21**	34	42

4.2.2. PETS 2009 dataset

Table 1 summarizes the GLMB-IM filter tracking performance against state-of-the-art baseline batch-based tracking algorithms [41], [40], [5] with the *PETS 2009* dataset [33] which includes high crowd density scenarios. We use well-known MOT performance indices for *PETS 2009* dataset such as MOT accuracy (MOTA), MOT Precision (MOTP), the number of fragmentations (FM) and the number of identity switches (IDS). Table 1 shows the ratio of tracks with successfully tracked parts for more than 80% (mostly tracked (MT)), less than 20% (mostly lost (ML)), or less than 80% and more than 20% (partially tracked (PT)). The up (down) arrows in Table 1 mean that higher (lower) the values indicate better performance. The best scores for individual attributes are marked as bold-faced. In the comparison results, we refer our algorithm as GLMB filter with image measurement (GLMB-IM). We include the GLMB filter [8] as the baseline to illustrate performance gain. As can be seen from the evaluation metrics, the performance of the GLMB-IM is similar to that of offline-based methods. A possible explanation for observing more ID switches and fragments is the re-initialization of tracks following a long-term full occlusion due to the measurement driven birth model.

4.2.3. MOT Challenge

In this section we test the GLMB-IM filter on the widely adopted *MOTChallenge* benchmark dataset [34] for more comprehensive comparisons. In this experiment, we also report false positives per frame (FPF), the number of false positives (FP), the number of false negatives (FN), and tracking speed in Hertz (Hz).

As can be seen from Table 2, the GLMB-IM filter achieves the best or second best performance in important indicators such as FPF, Recall and MT, amongst the online methods. For Frag and IDS, the GLMB-IM filter is consistently in the top three performers. More fragmentation is observed due to re-initialization of objects from the measurement-driven birth model when they emerge from very long full occlusions. Due to the generality of the framework, more sophisticated motion models and other types of detections and appearance features can be incorporated for further improvements. Selected frames from the tracking results for object occlusions are shown in Figure 9 where upper-left markers indicate object detections. Note from the tracking results without detections that the GLMB-IM tracker does not lose tracks due to false negatives or mutual occlusions. More surprisingly, it has comparable accuracy with batch methods, keeping in mind that it runs in near real-time with basic MATLAB implementation (see tracking speed in Hz).

The tracking experiments with the proposed GLMB-IM fil-

Table 2: Tracking performance on the 2D MOT dataset (Online methods are indicated by $*$)

Tracker	MOTA ↑	MOTP ↑	FAF ↓	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	FM ↓	Hz ↑
GLMB-IM (Ours) $*$	29.5	71.0	1.9	13.2%	40.1%	10,566	**32, 020**	670	1,260	20
TC_ODAL [42]$*$	15.1	70.5	2.2	3.2%	55.8%	12,970	38,538	637	1,716	1.7
MDP_REL [43]$*$	30.3	71.3	1.7	13.0%	**38.4%**	9,717	32,422	680	1,500	1.1
RMOT [44]$*$	18.6	69.6	2.2	5.3%	53.3%	12,473	36,835	684	1,282	7.9
DP_NMS [45]	14.5	70.8	2.3	6%	40.8%	13,171	34,814	4,537	3.090	**444.8**
TBD [46]	15.9	70.9	2.6	6.4%	47.9%	14,943	34,777	1,939	1,963	0.7
SMOT [47]	18.2	71.2	1.5	2.8%	54.8%	8,780	40,310	1,148	2,132	2.7
CEM [5]	19.3	70.7	2.5	8.5%	46.5%	14,180	34,591	813	1,023	1.1
SegTrack [48]	22.5	71.7	**1.4**	5.8%	63.9%	**7, 890**	39,020	697	**737**	0.2
MoiCon [49]	23.1	70.9	1.8	4.7%	52.0%	10,404	35,844	1,018	1,061	1.4
MHT_AM [40]	**32.4**	**71.8**	1.6	**16.0%**	43.8%	9,064	32,060	**435**	826	0.7

ter are implemented in MATLAB using single core (Intel i7 2.4GHz 5500) CPU laptop. A comparison of tracking speed with other trackers (excluding the point detection process) is summarized in Table 2, which shows an average of 20 fps for the GLMB-IM filter (without code optimization). Hence, the GLMB-IM filter is very well-suited for online applications considering further speed up can be achieved using C++ and code optimization. Further, the salient feature of the proposed GLMB-IM filter is its linear complexity with respect to the number of detections [28]. It is important to note that the reported computation speeds in Table 2 only serves as a rough indication because all implementations are dependent on the hardware platform, programming language, code structure, test sequence scenarios, etc.

In summary the GLMB-IM filter offers practical trade-offs between accuracy and speed for real-time applications. Further, as briefly mentioned before, the GLMB-IM filters can be extended to offline methods such as batch estimation or via smoothing techniques. The RFS approach also provides the probability distribution of the current number of objects, i.e., cardinality distribution Eq. (13) (which is not available in other tracking approaches). Figure 10 shows the frame by frame cardinality distribution for the three selected sequences.

5. Conclusion

This paper proposed an efficient online visual MOT algorithm that exploits the advantages of both detection-based and TBD approaches, which seamlessly integrates state estimation, track management, clutter rejection, false negatives and occlusion handling into one single Bayesian recursion. In particular, it has the efficiency of the detection-based approach that avoids updating with the entire image, while at the same time making use of information at the image level by using only small regions of the image where mis-detected objects are expected. The proposed algorithm has a linear complexity in the number of detections and quadratic in the number of hypothesized tracks, making it suitable for real-time computation. Experimental results on well-known datasets show that the proposed algorithm is ranked first or second amongst state-of-the-arts online methods, in terms of: the percentage of correctly tracked objects, percentage of tracks with successfully tracked parts; and the least false positive rates, and consistently in the top three performers in other standard indicators. More surprisingly, it has comparable accuracy with offline methods, keeping in mind that it runs in near real-time with basic Matlab implementation.

While the proposed MOT filter was developed for simple scenarios where tracks are terminated once they exit the field of view, the framework can accommodate more complex applications that require re-establishing identities of reappearing objects. In this case the state vector of each object is extended to incorporate appearance features, and their survival probabilities are set to unity once they reach a certain threshold (so that the filter will not terminate their tracks). Appearance models for each track can be learned on-line (via deep learning or other approaches) using the object's appearance features and probability of existence. When an object is detected in the field of view(s), the filter accounts for whether it is a new object or a reappearing object by updating the weights of relevant GLMB components/hypotheses in accordance to how well the appearance features of newly detected object fit the appearance models of the undetected objects. The major consideration in this extension is the computational cost due to the ever growing number of tracks, the increase in the dimension of the state of each object and the appearance learning process. Further work is needed to reduce the computational cost for real-time applications.

Acknowledgement

This work was supported by the Australian Research Council through a research grant DP160104662 and the National Strategic Project-Fine particle of the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT(MSIT), the Ministry of Environment(ME), and the Ministry of Health and Welfare(MOHW) (NRF-2017M3D8A1092022).

Figure 9: Selected frames of the tracking results for missing detections and mutual occlusions (Tracks are displayed by bounding boxes with ID (numbers) and detections are marked as upper-left corner markers.)

[1] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. V. Gool, Online multiperson tracking-by-detection from a single, uncalibrated camera, PAMI 33 (9) (2011) 1820–1833.

[2] K. Okuma, A. Taleghani, N. D. Freitas, D. G. Lowe, A boosted particle filter: Multitarget detection and tracking, in: Proc. Eur. Conf. Comput. Vis., 2004, pp. 28–39.

[3] L. Zhang, Y. Li, R. Nevatia, Global data association for multi-object tracking using network flows, in: CVPR, 2008, pp. 1–8.

[4] J. Berclaz, F. Fleuret, E. Turetken, P. Fua, Multiple object tracking using k-shortest paths optimization, PAMI 33 (9) (2011) 1806–1819.

[5] A. Milan, S. Roth, K. Schinlder, Continuous energy minimization for multitarget tracking, PAMI 36 (1) (2014) 58–72.

[6] S. Zhang, J. Wang, Z. Wang, Y. Gong, Y. Liu, Multi-target tracking by learning local-to-global trajectory models, Pattern Recognition 48 (2015) 580–590.

[7] A. Dehghan, Y. Tian, P. H. S. Torr, M. Shah, Target identity-aware network flow for online multiple target tracking, in: CVPR, 2015, pp. 1146–1154.

[8] B.-T. Vo, B.-N. Vo, Labeled random finite sets and multi-object conjugate priors, IEEE Trans. Signal Process. 61 (13) (2013) 3460–3475.

[9] S. Davey, M. Rutten, N. Gordon, Track-before-detect techniques, in: Integrated Tracking, Classification, and Sensor Management: Theory and Applications, Wiely/IEEE, New York, NY, USA, 2012, Ch. 8, pp. 311–361.

[10] B.-N. Vo, B.-T. Vo, N.-T. Pham, D. Suter, Joint detection and estimation of multiple objects from image observations, IEEE Trans. Signal Process. 58 (10) (2010) 5129–5241.

[11] F. Papi, D. Y. Kim, A particle multi-target tracking for superposional measurements using labeled random finite sets, IEEE Trans. Signal Process. 63 (16) (2015) 4348–4358.

[12] F. Papi, B.-N. Vo, B.-T. Vo, C. Fantacci, M. Beard, Generalized labeled multi-bernoulli approximation of multi-object densities, IEEE Trans. Signal Process. 63 (20) (2015) 5487–5497.

[13] M. Isard, J. MacCormick, Bramble: A bayesian multiple-blob tracker, in: Proc. Int. Conf. Comput. Vis., 2001, pp. 34–41.

[14] P. Pérez, C. Hue, J. Vermaak, M. Gangnet, Color-based probabilistic tracking, in: Proc. Eur. Conf. Comput. Vis., 2002, pp. 661–675.

[15] K. Nummiaro, E. Koller-Meier, L. V. Gool, An adaptive color-based particle filter, Image and Vision Computing 21 (10) (2003) 99–110.

[16] A. D. J. Vermaak, P. Pérez, Maintaining multi-modality through mixture tracking, in: Proc. Int. Conf. Comput. Vis., 2003, pp. 13–16.

[17] R. Hoseinnezhad, B.-N. Vo, B.-T. Vo, D. Suter, Visual tracking of numer-

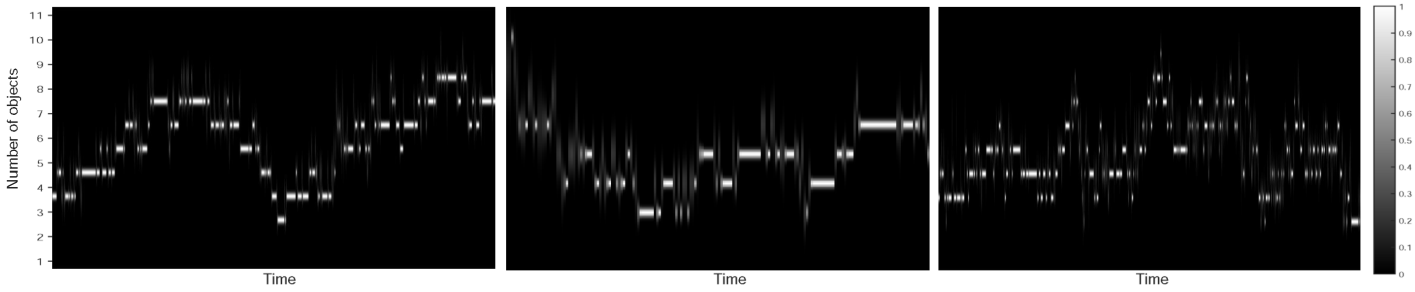

Figure 10: Example images of cardinality distributions (the probability distribution of the number of objects over time) for three sequences in *MOTChallenge* dataset (Left:*S2L1*, middle:*TUD*, Right:*ETH*)

ous targets via multi-bernoulli filtering of image data, Pattern Recognition 45 (10) (2012) 3625–3635.

[18] R. Mahler, Statistical Multisource-Multitarget Information Fusion, Artech House, Inc.e, Norwood, MA, USA, 2007.

[19] S. S. Intille, A. F. Bobick, Closed-world tracking, in: Proc. Int. Conf. Comput. Vis., 1995, pp. 672–678.

[20] M. Kristan, J. Perš, M. Perš, S. Kovačič, Closed-world tracking of multiple interacting targets for indoor-sports applications, Computer Vision and Image Understanding 113 (5) (2009) 598–611.

[21] R. Mahler, Multitarget bayes filtering via first-order multitarget moments, IEEE Trans. Aerosp. Electron. Sys. 39 (4) (2003) 1152–1178.

[22] R. Mahler, Advances in Statistical Multisource-Multitarget Information Fusion, Artech House, Inc.e, Norwood, MA, USA, 2014.

[23] B.-N. Vo, S. Singh, A. Doucet, Sequential monte carlo methods for multitarget filtering with random finite sets, IEEE Trans. Aerosp. Electron. Sys. 41 (5) (2005) 1224–1245.

[24] B.-N. Vo, B.-T. Vo, N.-T. Pham, D. Suter, A particle marginal metropolis-hastings muti-targettracker, IEEE Trans. Signal Process. 62 (15) (2014) 3953–3964.

[25] P. Craciun, M. Ortner, J. Zerubia, Joint detection and tracking of moving objects using spatio-temporal marked point processes, in: IEEE Winter Conf. on Applications of Computer Vision, 2015, pp. 177–184.

[26] B.-T. Vo, B.-N. Vo, D. Phung, Labeled random finite sets and the bayes multi-target tracking filter, IEEE Trans. Signal Process. 62 (24) (2014) 6554–6567.

[27] M. Beard, B.-T. Vo, B.-N. Vo, K. Dietmayer, Bayesian multi-target tracking with merged measurements using labelled random finite sets, IEEE Trans. Signal Process. 63 (6) (2015) 1433–1447.

[28] B.-N. Vo, B.-T. Vo, H. G. Hoang, An efficient implementation of the generalized labeled multi-bernoulli filter, IEEE Trans. Signal Process. 65 (8) (2017) 1975–1987.

[29] S. Reuter, B.-T. Vo, B.-N. Vo, K. Dietmayer, The labeled multi-bernoulli filter, IEEE Trans. Signal Process. 62 (12) (2014) 3246–3260.

[30] J. F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, PAMI 37 (3) (2015) 583–596.

[31] X. Shen, X. Sui, K. Pan, T. Y., Adaptive pedestrian tracking via patch-based features and spatial-temporal similarity measurement, Pattern Recognition 53 (2016) 163–173.

[32] D. Schuhmacher, B.-T. Vo, B.-N. Vo, A consistent metric for performance evaluation of multi-object filters, IEEE Trans. Signal Process. 56 (8) (2008) 3447–3457.

[33] J. Ferryman, in: IEEE Workshop Performance Evaluation of Tracking and Surveillance, 2009.

[34] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, K. Schindler, Motchallenge 2015: Towards a benchmark for multi-target tracking, arXiv:1504.01942 [cs]ArXiv: 1504.01942.

[35] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part based models, PAMI 36 (1) (2014) 58–72.

[36] R. Mahler, B.-T. Vo, B.-N. Vo, CPHD filtering with unknown clutter rate and detection profile, IEEE Trans. Signal Process. 59 (8) (2011) 3497–3513.

[37] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Q. Tian, Person re-identification in the wild, in: CVPR, 2017, pp. 3346–3355.

[38] B.-N. Vo, B.-T. Vo, R. Mahler, Closed-form solutions to forward-backward smoothing, IEEE Trans. Signal Process. 60 (1) (2011) 2–17.

[39] M. Beard, B.-T. Vo, B.-N. Vo, Generalized labelled multi-bernoulli forward-backward smoothing, in: Proc. Int' Conf. Inf. Fusion, 2016, pp. 688–694.

[40] C. Kim, F. Li, A. Ciptadi, J. M. Rehg, Multiple hypothesis tracking revisited, in: ICCV, 2015.

[41] I. J. Cox, S. L. Hingorani, An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking, PAMI 18 (2) (1996) 138–150.

[42] S. Bae, K. Yoon, Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning, in: CVPR, 2014.

[43] Y. Xiang, A. Alahi, S. Savarase, Learning to track: Online multi-object tracking by decision making, in: ICCV, 2015.

[44] J. H. Yoon, M. H. Yang, J. Lim, K.-J. Yoon, Bayesian multi-object tracking using motion context from multiple objects, in: IEEE Winter Conf. on App. Comput. Vis., 2015, pp. 33–40.

[45] H. Pirsiavash, D. Ramanan, C. C. Fowlkes, Globally optimal greedy algorithms for tracking a variable number of objects, in: CVPR, 2011.

[46] A. Geiger, M. Lauer, C. Wojek, C. Stiller, R. Urtasun, 3d traffic scene understanding from movable platform, PAMI 36 (5) (2014) 1012–1025.

[47] C. Dicle, O. Camps, M. Sznaier, The way they move: Tracking targets with similar appearance, in: ICCV, 2013.

[48] A. Milan, L. Leal-Taixe, I. Reid, K. Schindler, Joint tracking and segmentation of multiple targets, in: CVPR, 2015.

[49] L. Leal-Taixe, M. Fenzi, A. Kuznetsova, B. Rosenhahn, S. Savarese, Learning an image-based motion context for multiple people tracking, in: CVPR, 2014.