Learning scene context for multiple object tracking

Emilio Maggio*, Andrea Cavallaro

***Abstract*—We propose a framework for multi-target tracking with feedback that accounts for scene contextual information. We demonstrate the framework on two types of context-dependent events, namely target births (i.e., objects entering the scene or reappearing after occlusion) and spatially persistent clutter. The spatial distributions of birth and clutter events are incrementally learned based on mixtures of Gaussians. The corresponding models are used by a Probability Hypothesis Density (PHD) filter that spatially modulates its strength based on the learned contextual information. Experimental results on a large video surveillance dataset using a standard evaluation protocol show that the feedback improves the tracking accuracy from 9% to 14% by reducing the number of false detections and false trajectories. This performance improvement is achieved without increasing the computational complexity of the tracker.**

***Index Terms*—Adaptive filtering, video surveillance, clutter, tracking, GMM, context, PHD filter.**

I. Introduction

IMAGE-BASED trackers may fail in real world scenarios due to the performance limitations of object detectors that generate noisy observations under illumination changes, reflections and occlusions. Temporal filtering is usually applied to cope with errors due to these uncertain observations. As the performance of a detector depends on the scene characteristics (such as object–background separability, areas of occlusions, entry and exit points and dynamic textures), additional information on scene context may help the tracker disambiguate real targets from clutter. Although the use of low-level contextual information has been investigated for improving the performance of the detector itself [1], [2], the use of contextual information for improving the spatio-temporal filtering performance is still an open issue.

Bayesian recursion is a popular approach to filter noisy observations in target tracking [3], [4]. The Bayes filter first predicts the target state based on a dynamical model and then updates the resulting distribution using new observations. Unlike single-target Bayes trackers, which only remove spatial noise from the input data, multi-target trackers must account for target birth and target death, clutter and missing observations, and ideally smoothing the input both in space and time.

A popular solution to deal with clutter and missing observations is the Multiple Hypothesis Tracker (MHT) [5], [6]. MHT explicitly postulates multiple association hypotheses between the set of observations and a finite set of multi-target hypothetical states; this allows one to correct past estimates with future data. The match between a detection and a trajectory in a particular association hypothesis is computed using a Kalman Filter. However, as the number of association hypotheses grows exponentially with the time and with the number of targets, a gating procedure is necessary to discard less promising associations and to reduce the number of Kalman filter computations. A similar path is followed in [7] where the marginal association *pdf*s of the Joint Probability Data Association Filter (JPDAF) are sampled using a Particle Filter (PF). The approach is less complex than sampling the full multi-target state, as filtering is applied to independent association hypotheses. The data association problem can also be modelled in a deterministic framework using graph theory [8]. The graph structure accounts for target birth, death and missing detections, but a pre-filtering step is necessary to remove spatial noise and clutter. A graph based method can solve the association problem in a multi-sensor setup [9]. In this case data association is performed across time, space and multiple views. Jump Markov Systems (JMS) approximated by PF have also been used to model a time-varying number of targets in the scene, clutter and missing detections [10], [11]. The JMS models the dependencies between targets and allows for efficient design of the importance sampling function. Recently, Rao-Blackwellization (RB) has been used to reduce the computational cost of a multi-target Monte Carlo filter [12]. This filter integrates the state propagation in closed form, while Monte Carlo integration is used for data association. Also, in visual tracking the one-to one assumption made by most of the data association algorithms does not hold as the image region can be split into multiple blobs by the background subtraction algorithm. A Markov Chain Monte Carlo tracker can solve this problem by propagating both in space and time the association hypotheses [13]. However, none of the trackers described so far is a natural extension of the single-target Bayes recursion to multi-target tracking. Trackers like MHT [5] and JPADF [7] apply independent Bayes filtering to each association hypothesis and not to the multi-target state, thus reducing the filtering problem to a single-target one. In these filters the estimate of the current number of targets is a consequence of the selection of the best association hypothesis.

Recently Mahler proposed a new formulation of the multi-target tracking problem and of the Bayes recursion which makes use of Finite Set Statistics (FISS) [14]. This framework considers the multi-target state as a single meta-target and the

This work was done when E. Maggio was with Queen Mary, University of London. He is now with VICON (Oxford Metric Group), United Kingdom e-mail: emilio.maggio@vicon.com. A. Cavallaro is with the Multimedia and Vision Group - Queen Mary, University of London, United Kingdom, E1 4NS, UK e-mail: andrea.cavallaro@elec.qmul.ac.uk. The authors acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC), under grant EP/D033772/1.

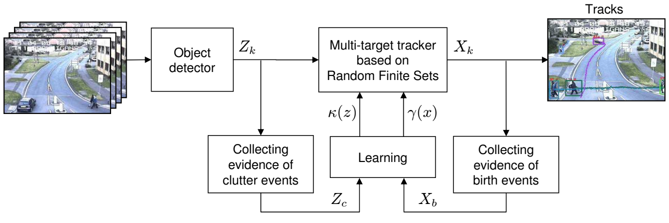

Tracks
Object detector
Z_k
Multi-target tracker based on Random Finite Sets
X_k
$\kappa(z)$
$\gamma(x)$
Collecting evidence of clutter events
Learning
Collecting evidence of birth events
Z_c
X_b

Fig. 1. Feedback schema for target tracking based on object detection and on Particle PHD filtering. The output (Z_k) gathered by the detector and by the tracker (X_k) is filtered and used to extract contextual information in the form of statistics on the detector failure modalities ($\kappa(z)$) and on the object entry areas in the scene ($\gamma(x)$). The PHD filter uses this feedback to modulate the filtering strength and improve the tracking performance.

observations as a single set of measurements of the meta-sensor [15]. As the dimensionality of the target state grows exponentially with the number of targets, the Probability Hypothesis Density (PHD) filter can be used as a computationally efficient algorithm to propagate the first-order moment of the multi-target statistics only [14]. The PHD filter maintains the same modeling capabilities of the full Bayesian multi-target recursion in terms of clutter, birth and missing observations but with linear complexity with the number of targets in the scene. Experimental results have shown that a tracking pipeline based on the PHD filter achieves improved performance with respect to standard methods like MHT [15].

Also, A tracker based on the PHD filter can use scene contextual information by allowing state-space-dependent models of *birth* and *clutter*. As considering spatial dependencies requires the introduction of additional parameters in the filter, the complexity is usually limited either by working with synthetic data, whose model matches exactly that used in the filter [15], or by using uniform distributions [16], [17], [18], [19]. The latter case leads to a filter that is independent from the scene context and does not exploit the full capabilities of the RFS tracking framework.

II. Contribution

THE research presented in this paper builds on our previous work on multi-target tracking with the particle PHD filter [20], [21], where we showed that a tracker based on the Particle PHD filter outperforms in visual scenarios a classic graph-based multiple-hypotheses data association method [8].

In this paper instead we investigate the use of scene contextual information to improve the accuracy of the overall tracking result. First, we combine automatic and interactive feedback to extract scene contextual information (Fig. 1). Then we use the natural modeling capabilities of the Bayesian multi-target tracking framework to locate where objects are more likely to appear in the scene (*birth events*) and where the detector is expected to produce errors (*clutter events*). To this end, we propose to model birth and clutter data using a parametric model (GMM) learned incrementally [22]. We then use these models in the Bayesian tracker based on the PHD filter recursion to modulate the filter response depending on the location of the candidate targets. We demonstrate this framework using background–subtraction–based tracking [23] and evaluate the results on a large outdoor surveillance dataset (the CLEAR-2007 dataset).

The paper is organized as follows. Section III introduces the multi-target tracking framework based on the Particle PHD filter. Section IV describes the density estimation technique used to enable the inclusion of contextual information in the tracker. In Section V we discuss the results on a standard dataset and in Section VI we draw conclusions.

III. The multi-target tracking framework

THIS section offers an overview of our visual tracker based on Random Finite Sets (RFS) and the PHD filter. A more detailed explanation of the RFS tracking theory and a detailed description of our implementation are presented in [15] and [21], respectively.

A. Random Finite Sets for multi-target tracking

Let the target area be approximated with a $w \times h$ rectangle centered in $\left(y^{(1)}, y^{(2)}\right)$. Let the single-target state at time k be $x_k = (y^{(1)}_{x_k}, \dot{y}^{(1)}_{x_k}, y^{(2)}_{x_k}, \dot{y}^{(2)}_{x_k}, w_{x_k}, h_{x_k}) \in E_s$, where $(\dot{y}^{(1)}_{x_k}, \dot{y}^{(2)}_{x_k})$ is the speed of the target and E_s is the single-target space. Finally, let the single-target observation $z_k = (y^{(1)}_{z_k}, y^{(2)}_{z_k}, w_{z_k}, h_{z_k}) \in E_o$ be a rectangle generated by an object detector, with E_o representing the observation space. We then define the multi-target state, X_k, and measurement, Z_k, as the finite collection of the states and observations of each target. If $M(k)$ is the number of visible targets at time k, then the multi-target state, X_k, is the set

$$X_k = \left\{x_{k,1}, ...x_{k,M(k)}\right\} \in \mathcal{F}(E_s), \tag{1}$$

where $\mathcal{F}(E)$ is the collection of all the finite subsets of E. The multi-target measurement, Z_k, is the set

$$Z_k = \left\{z_{k,1}, ...z_{k,N(k)}\right\} \in \mathcal{F}(E_o), \tag{2}$$

which is formed by the $N(k)$ observations. Note that some observations may be due to clutter and some targets may fail to generate observations.

The uncertainty in the state and measurement is introduced in the framework of finite sets statistic by modeling the multi-target state and the multi-target measurement using two Random Finite Sets (RFS). An RFS is a finite set of random vectors for which the cardinality is also a random variable. Let Ξ_k be the RFS associated with the multi-target state:

$$\Xi_k = S_k\left(X_{k-1}\right) \cup B_k\left(X_{k-1}\right) \cup \Gamma_k, \tag{3}$$

where $S_k\left(X_{k-1}\right)$ denotes the RFS of survived targets, while $B_k\left(X_{k-1}\right)$ is the RFS of target spawned from the previous set of targets X_{k-1}, and Γ_k is the RFS of the new-born targets [15]. The RFS Ω_k associated with the measurement is defined as

$$\Omega_k = \Theta_k\left(X_k\right) \cup K_k, \tag{4}$$

where $\Theta_k\left(X_k\right)$ is the RFS of the measurements generated by the targets X_k, and K_k models clutter and false detections.

The goal is to estimate $p_{k|k}(X_k|Z_{1:k})$, the *pdf* of the objects being in state X_k given all the observations $Z_{1:k}$ up to time k, based on the previous two sets equations (Eq. (3) and Eq. (4)).

The estimation is performed recursively in two steps, namely prediction and update. The *prediction step* uses the dynamic model defined in Eq. (3) to obtain the prior *pdf* as

$$\begin{aligned} & p_{k|k-1}(X_k|Z_{1:k-1}) \\ & = \int f_{k|k-1}(X_k|X_{k-1}) p_{k-1|k-1}(X_{k-1}|Z_{1:k-1}) \mu(dX_{k-1}) \end{aligned} \tag{5}$$

with $p_{k-1|k-1}(X_{k-1}|Z_{1:k-1})$ known from the previous iteration and the transition density $f_{k|k-1}(X_k|X_{k-1})$ determined by Eq. (3). μ is an appropriate dominating measure on $\mathcal{F}(E_s)$ (for a detailed description of RFSs, set integrals and formulation of μ, please refer to [14] and [15]). The *update step* uses the Bayes' rule once the observation Z_k is available

$$p_{k|k}(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k) p_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X_k) p_{k|k-1}(X_k|Z_{1:k-1}) \mu(dX_k)}. \tag{6}$$

Although a Monte Carlo approximation of this recursion is possible [15], the number of particles required grows exponentially with the number of targets. Therefore, an approximation is necessary to make the problem computationally tractable. An example of approximation is the propagation of the first-order moment of the multi-target posterior only, instead of the posterior itself [14], as described in the next section.

B. The Particle PHD tracker

The Probability Hypothesis Density (PHD) is a function in the single-target state space whose peaks identify the likely position of the targets. The PHD, $\mathcal{D}_\Xi(x)$, is the first order moment of a RFS, Ξ, and is a function on E_s. The property of the PHD is that for any region $R \subseteq E_s$

$$E[|\Xi \cap R|] = \int_R \mathcal{D}_\Xi(x) dx, \tag{7}$$

where $|.|$ denotes the cardinality of a set. Eq. (7) means that by integrating the PHD on any region R of the state space we obtain the expected number of targets in R.

Let $\mathcal{D}_{k|k}(x)$ be the PHD at time k associated with the multi-target posterior density $p_{k|k}(X_k|Z_{1:k})$, then the Bayesian iterative prediction and update of $\mathcal{D}_{k|k}(x)$ is known as the PHD filter. Although no generic algebraic solution exists for the PHD filter integrals, a Monte Carlo solution that approximates the PHD with a (large) set of weighted random samples is possible (the Particle PHD filter [15]). Let the set $\{\omega_{k-1}^{(i)}, x_{k-1}^{(i)}\}_{i=1}^{L_{k-1}}$ of L_{k-1} particles with state $x_{k-1}^{(i)}$ and associated weight $\omega_{k-1}^{(i)}$ approximate the PHD at time $k-1$. In this case the densities $p_{k|k}(x_k|z_{1:k})$ are approximated with a sum of L Dirac functions (i.e., particles) centered in $\{x_k^{(i)}\}_{i=1}^{L}$:

$$\mathcal{D}_{k-1|k-1}(x) \approx \sum_{i=1}^{L_{k-1}} \omega_{k-1}^{(i)} \delta\left(x - x_{k-1}^{(i)}\right). \tag{8}$$

Let a new set of particles $\{\tilde{\omega}_k^{(i)}, \tilde{x}_k^{(i)}\}_{i=1}^{L_{k-1}+J_k}$ be generated by drawing L_{k-1} samples from the importance function $q_k(.|x_{k-1}^{(i)}, Z_k)$. These samples propagate the tracking hypotheses from the samples at time $k-1$. Then J_k samples are drawn from the new-born importance function, $p_k(.|Z_k)$, representing the state hypotheses of new targets appearing in the scene. The predicted weights, $\tilde{\omega}_{k|k-1}^{(i)}$, are defined as

$$\tilde{\omega}_{k|k-1}^{(i)} = \begin{cases} \frac{\phi_{k|k-1}\left(\tilde{x}_k^{(i)}, x_{k-1}^{(i)}\right) \omega_{k-1}^{(i)}}{q_k\left(\tilde{x}_k^{(i)} | x_{k-1}^{(i)}, Z_k\right)} & i = 1, ..., L_{k-1} \\ \frac{\gamma_k(\tilde{x}_k^{(i)})}{J_k p_k\left(\tilde{x}_k^{(i)} | Z_k\right)} & i = L_{k-1}+1, \ldots, L_{k-1}+J_k \end{cases}. \tag{9}$$

where $\gamma_k(.)$ is the intensity function of the new target birth RFS. The integral of $\gamma_k(.)$ over a region R gives the expected number of new objects per frame appearing in R. $\phi_{k|k-1}(x, \xi)$ is the pseudo-state transition probability $\phi_{k|k-1}(x, \xi) = e_{k|k-1}(\xi) f_{k|k-1}(x|\xi)$, where $e_{k|k-1}(\xi)$ is the probability that the target still exists at time k and $f_{k|k-1}(x|\xi)$ is the single-target state transition probability.

Once the new set of observations is available, the weights $\{\tilde{\omega}_{k|k-1}^{(i)}\}_{i=1}^{L_{k-1}+J_k}$ are updated according to

$$\tilde{\omega}_k^{(i)} = \left[p_M(\tilde{x}_k^{(i)}) + \sum_{z \in Z_k} \frac{\psi_{k,z}(\tilde{x}_k^{(i)})}{\kappa_k(z) + C_k(z)} \right] \tilde{\omega}_{k|k-1}^{(i)}, \tag{10}$$

where $C_k(z) = \sum_{j=1}^{L_{k-1}+J_k} \psi_{k,z}(\tilde{x}_k^{(i)}) \omega_{k|k-1}^{(j)}$, $p_M(x)$ is the missing detection probability; $\psi_{k,z}(x) = (1-p_M(x)) g_k(z|x)$, $g_k(z|x)$ is the single-target likelihood defining the probability that z is generated by a target with state x and $\kappa_k(.)$ is the clutter intensity.

As the bigger an object in the image, the larger its acceleration, we correlate the change of state with the target size. Thus the state transition $f_{k|k-1}(x_k|x_{k-1})$ is a first-order Gaussian dynamic with State Dependent Variances (SDV), that is

$$x_k = \overbrace{\left[\begin{array}{ccc} A & 0_2 & 0_2 \\ 0_2 & A & 0_2 \\ 0_2 & 0_2 & I_2 \end{array}\right]}^{G} x_{k-1} + \left[\begin{array}{cc} B_1 & 0_2 \\ B_2 & 0_2 \\ 0_2 & B_3 \end{array}\right] \left[\begin{array}{c} n_k^{(1)} \\ n_k^{(2)} \\ n_k^{(w)} \\ n_k^{(h)} \end{array}\right], \tag{11}$$

with

$$A = \left[\begin{array}{cc} 1 & T \\ 0 & 1 \end{array}\right], \; B_1 = w_{x_{k-1}} \left[\begin{array}{cc} \frac{T^2}{2} & 0 \\ T & 0 \end{array}\right],$$

$$B_2 = h_{x_{k-1}} \left[\begin{array}{cc} 0 & \frac{T^2}{2} \\ 0 & T \end{array}\right], \;\; and \;\; B_3 = \left[\begin{array}{cc} T w_{x_{k-1}} & 0 \\ 0 & T h_{x_{k-1}} \end{array}\right],$$

where 0_n and I_n are the $n \times n$ zero and identity matrices, and $\{n_k^{(1)}\}$,$\{n_k^{(2)}\}$, $\{n_k^{(w)}\}$ and $\{n_k^{(h)}\}$ are independent white Gaussian noises with standard deviations $\sigma_{n^{(1)}}$,$\sigma_{n^{(2)}}$,$\sigma_{n^{(w)}}$ and $\sigma_{n^{(h)}}$, respectively. $T = 1$ is the interval between two consecutive steps, $k-1$ and k.

Similarly to the state transition case, we correlate the amplitude of the measurement noise to the target size. Thus, we define the single-target likelihood $g_k(z|x)$ as a Gaussian with SDV, such that $g_k(z|x) = \mathcal{N}(z; Cx, \Sigma(x))$, where $\mathcal{N}(.)$ is a Gaussian evaluated in z, centered in Cx and with covariance matrix $\Sigma(x)$. C is defined as

$$C = \left[\begin{array}{cc} D & 0_{2\times 3} \\ 0_{2\times 4} & I_2 \end{array}\right], \quad \text{with } D = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 0 & 1 \end{array}\right]$$

and $0_{n\times m}$ is the $n \times m$ zero matrix. $\Sigma(x)$ is diagonal where $diag(\Sigma(x)) = [\frac{\sigma_{v^{(w)}}}{2} w_x, \frac{\sigma_{v^{(h)}}}{2} h_x, \sigma_{v^{(w)}} w_x, \sigma_{v^{(h)}} h_x]$.

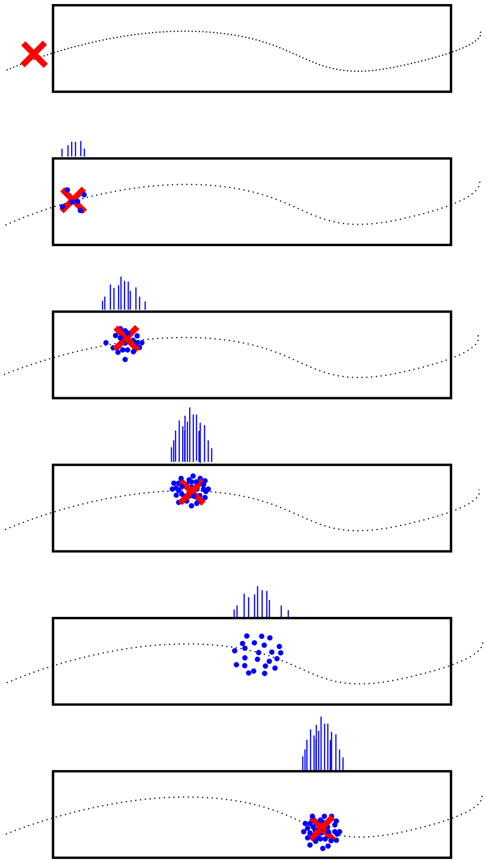


Fig. 2. Illustration of the Particle PHD filter. From top to bottom: a target appears in the camera visual field (black box). A set of new particles (blue dots) grows around the measurements (red crosses). The cumulative mass grows. In the case of missing detections, the particles keep on propagating according to the state transition and eventually join the measurement.

In the absence of any prior knowledge, the missing detection probability, $p_M(x)$, the probability of survival, $e_{k|k-1}(x)$, the birth and clutter intensities $\gamma_k(x)$ and $\kappa_k(z)$ are constant over x and z. Sec. IV will show how to generate more complex forms for $\gamma_k(x)$ and $\kappa_k(z)$ to model contextual information.

At each iteration, J_k new particles are added to the old L_{k-1} particles. To limit the growth of the number of particles, a resampling step is performed after the update step. The final set of particles with associated weights $\{\omega_k^{(i)}, x_k^{(i)}\}_{i=1}^{L_k}$ representing the PHD is defined in the single-target state space and is normalized to preserve the total mass. Figure 2 shows a pictorial representation of the particle PHD propagation in the *single target case*. When the target appears in the camera visual field, a set of new born particles grows around the measurements. If the series of measurements is deemed plausible by the state transition and observation models, then the mass of the particles grows towards 1. When the target does not generate a measurement, then the number of resampled particles decreases. If after few steps the target reappears, then the surviving particles form again the cluster around the measurement. Otherwise, the number of particles goes to zero, thus indicating the absence of a target. In the *multi-target case*, the set of particles also carries information about the expected *number* of targets in the scene. An example of PHD approximated by particles is shown in Fig. 3. The peaks of the PHD are on the detected vehicles, and the mass $\hat{M}_{k|k} \approx 3$ estimates the number of targets.

As the PHD does not hold any information about the identity of the targets, we first use clustering with Gaussian Mixture Models (GMM) to detect the peaks of the PHD. To filter out the data due to clutter, after GMM, we select as candidate states the centers of the clusters whose mass is at greater than 0.5. The resulting candidate states validated over space and time by the PHD filter are processed by a graph matching procedure for data association [8], [21], which consistently links the previous candidate states with the new ones thus propagating the target identity (Fig. 3 (c)).

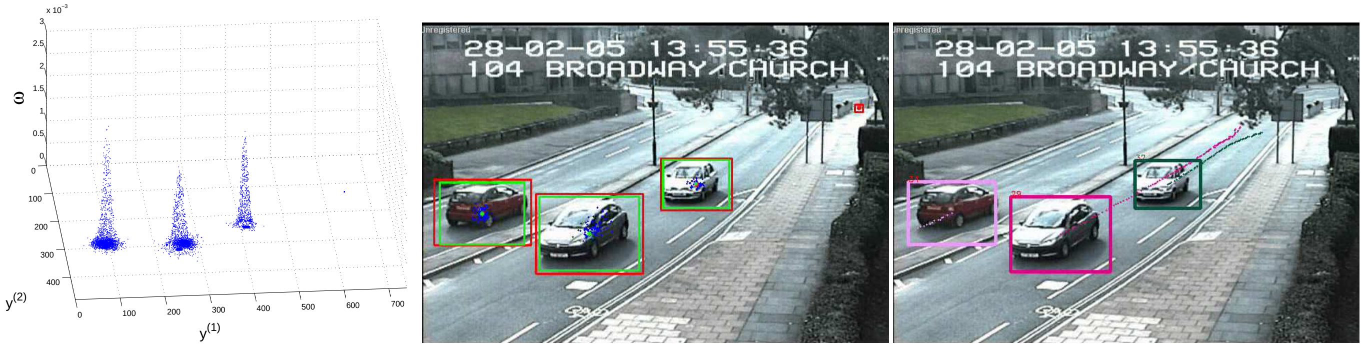
IV. Learning birth and clutter intensity

A. Contextual information

THE propagation model defined in Eq. (9) and Eq. (10) offers several degrees of freedom that help tuning the filtering behavior depending on the tracking scenario at hand. Recent work on semantic region modeling has shown that we can extract contextual information of the scene using the output of a tracker and a parametric model [24], [25]. Common activity paths, object entry and exit areas are detected and used for higher level behavioural analysis. Similarly, in this section we detail the procedure to learn a parametric model of the birth and clutter intensities $\gamma_k(x)$ and $\kappa_k(z)$ that introduces in the tracker scene contextual knowledge.

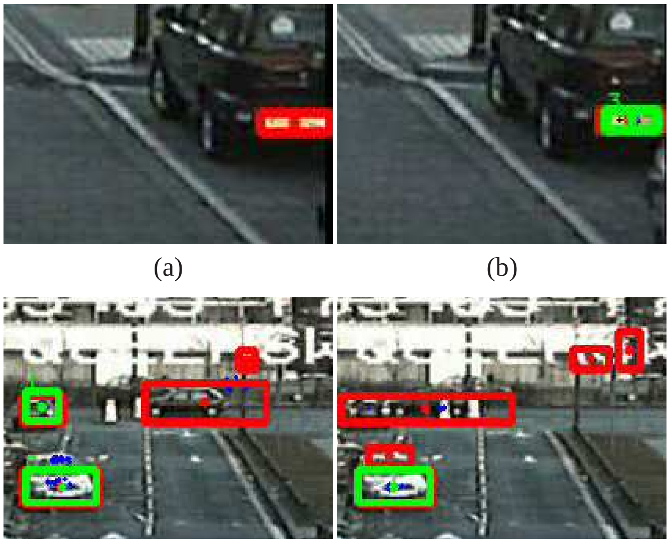
A simple solution to model clutter and birth intensities is to consider $\gamma_k(x)$ and $\kappa_k(z)$ uniform on x and z, respectively. In this case, the absolute values of the intensities representing the average number of birth and clutter events per frame are the only parameters to choose. However, as this solution requires a compromise between the various image regions, it may produce sub-optimal results. Fig. 4 (a)-(b) shows an example of PHD-based tracker results where the filtering is too weak. Spatially consistent detections caused by an illumination change are produced by the number plate of the car. Although this object is in a position where a new target is unlikely to be born, the filtering effect of the PHD is not strong enough and the tracker generates a false track. Fig. 4 (c)-(d) shows an example of results where filtering is too strong. A car in the far field is detected for few frames. Although the probability of appearance of a vehicle is high in that image region, the PHD filters out the detections, thus loosing the track.

Fig. 5 (a) shows an example of image areas where new objects are likely to appear. Hereby, the target birth model should account for spatial variability in the scene and allow the filter to reduce temporal smoothing over these locations. Likewise, Fig. 5 (b) shows an example of image areas where a detector based on background subtraction is expected to fail (on the white lines due to reflections and illumination changes and on waving vegetation). Therefore, the density of a birth and clutter event should depend on the scene contextual information. The PHD filter can account for scene context by varying its filtering strength according to the hypothetical state of a target. This is possible as the birth intensity $\gamma_k(x)$ may depend on the state x and the clutter intensity $\kappa_k(z)$



(a) (b) (c)

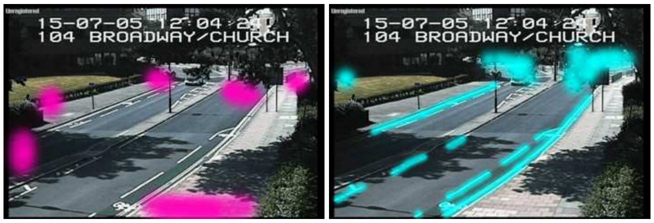
Fig. 3. (a) Example of particles approximating the PHD (before the resampling step) corresponding to the three visual targets of (b). The red boxes are the detections used as input, whereas the green boxes indicate the centers of each cluster. (c) Output trajectories after data association.



(a) (b)

(c) (d)

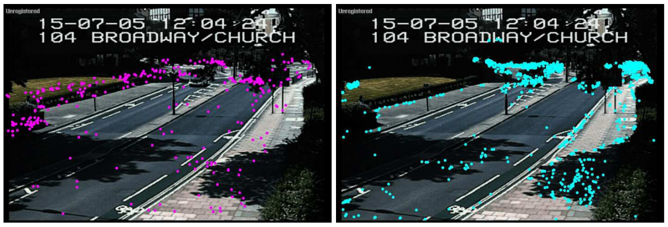
Fig. 4. Examples of failures of a PHD-based tracker that uses uniform birth and clutter intensities on the scenario S201 of the CLEAR-2007 dataset. The use of uniform densities leads to insufficient filtering that results in a false track on the number plate of the car (a)-(b) and on excessive filtering that results in missed target (black car) (c)-(d) (red: detections from a background subtraction algorithm; green: the PHD output).



(a) (b)

Fig. 5. Image areas where target birth events (a) and clutter events (b) are likely to happen.

may depend on the observation z. The acquisition of event and clutter information for intensity learning is based on the analysis of the output of the tracker and of the detector. In the following we describe how we learn non-uniform models of birth and clutter intensities.



(a) (b)

Fig. 6. Position of the centroids of the birth events (a) and of the clutter events (b) for scenarios S101 from the CLEAR-2007 dataset.

B. *Intensity learning*

Learning intensity functions reduces to a density estimation problem by decomposing the birth and clutter intensities, $\gamma_k(x)$ and $\kappa_k(z)$, as

$$\gamma_k(x) = \bar{s}_k p_k(x|b), \tag{12}$$

and

$$\kappa_k(z) = \bar{r}_k p_k(z|c), \tag{13}$$

where $\bar{s}_k$ and $\bar{r}_k$ are the average birth/clutter events per frame and $p_k(x|b)$ and $p_k(z|c)$ are the distributions in the state and observation spaces, respectively. For simplicity we assume that the two intensities are stationary and therefore we drop the temporal subscript k. We can include this information in the PHD filtering model by computing approximated versions of $\bar{s}$, $\bar{r}$, $p(x|e)$ and $p(z|c)$. The computation of $\bar{s}$ and $\bar{r}$ simply requires the cardinality of the sets and it is computed as the average over all the frames; $p(x|b)$ and $p(z|c)$ instead require density estimation in 6D or 4D spaces.

Fig. 6 (a) shows the birth event centroids obtained from the analysis of a 20-minute surveillance video clip. Clusters are mainly localized on the road and on the sidewalks. Birth events are also generated in non-entry areas because of track re-initializations due to object proximity and occlusions.

To approximate $p(x|b)$, we have to select a density estimation technique. Density estimation models can be classified into three main groups: parametric, non-parametric and semi-parametric models [26]. As the target birth probability is likely to be multi-modal with localized peaks on few image regions (i.e., a door, a road, etc.) a fully *Parametric model* that approximates the density with simple forms like a Gaussian is not appropriate. A *Non-parametric model* like Kernel Density Estimation (KDE) instead can represent complex distributions as it builds the estimate over the data. Unfortunately, KDE is potentially too demanding in terms of memory. In fact, the estimation requires the original dataset of birth events that potentially grows with the time. Consequently this method may not be appropriate for real-world surveillance systems where the tracker runs on board of a smart camera with limited resources. Therefore, to model $p(x|b)$ we use a *Semi-parametric method* where the sample joint-distribution is obtained with a combination (a mixture) of simple parametric models [26]. The parameters of the mixtures are usually learned by finding the Maximum Likelihood (ML) solution via a numerical method like Expectation-Maximization (EM) [27]. However, two problems arise with a classical ML-EM implementation. First, the likelihood is not a good indicator for model selection as it monotonically grows with the number of components. Second, if new data becomes available, EM requires again the complete dataset to update the model. A solution to the first problem is to impose a prior on the parameters which favors simpler models and substitute ML with a Maximum a Posteriori (MAP) solution [28], [29]. The second problem can be solved by using a recursive form for the MAP-EM equations [22], [30]. To model the distribution of birth events we use a modified version of this MAP estimate [22].

We approximate the distribution $p(x|b)$ with a mixture-of-Gaussian components that can be expressed as

$$p(x|b) \approx p(x|\theta) = \sum_{m=1}^{M} \pi_m p_m(x|\theta_m), \quad \text{with} \sum_{m=1}^{M} \pi_m = 1, \tag{14}$$

where $\theta = \{\pi_1, \ldots, \pi_M, \theta_1, \ldots, \theta_M\}$ is the set of parameters defining the mixture, M is the number of components and $p_m(x|\theta_m) = \mathcal{N}(x, \mu_m, \Sigma_m)$ is the m-th Gaussian component with parameters $\theta_m = \{\mu_m, \Sigma_m\}$, and μ_m and Σ_m are the mean and covariance, respectively. The goal is to find the optimal set θ_{MAP} that maximizes the log-posterior as

$$\theta_{MAP} = \arg\max_{\theta} \left\{ \log p(X_b|\theta) + \log p(\theta) \right\}. \tag{15}$$

Starting with a large number of components, the algorithm converges toward the MAP estimate for θ by selecting the number of components important for the estimation using the Dirichlet prior

$$p(\theta) \propto \prod_{m-1}^{M} \pi_m^{-\tau}, \tag{16}$$

where $\tau = N/2$, and N is the number of parameters per component in the mixture. In the Dirichlet distribution τ represents the prior evidence of a component. When τ is negative (i.e., improper Dirichlet) the prior allows for the existence of a component only if enough evidence is gathered from the data. The prior drives the irrelevant components to extinction, thus favoring simpler models.

Given the MAP estimate $\theta^{(n)}$ obtained using n data points $\{x^{(1)}, \ldots, x^{(n)}\}$ and the new data $x^{(n+1)}$, we obtain the updated estimate $\theta^{(n+1)}$ by first computing the ownerships

$$o_m^{(n)}(x^{(n+1)}) = \pi_m^{(n)} p_m(x^{(n+1)}|\theta_m^{(n)}) / p(x^{(n+1)}|\theta^{(n)}) \tag{17}$$

and by then updating the parameters as

$$\pi_m^{(n+1)} = \pi_m^{(n)} + \alpha \left(\frac{o_m^{(n)}(x^{(n+1)})}{1 - M\tau\alpha} - \pi_m^{(n)} - \frac{\tau\alpha}{1 - M\tau\alpha} \right), \tag{18}$$

for a Gaussian Mixture with $p_m(x|\theta_m) = N(x, \mu_m, \Sigma_m)$ then

$$\mu_m^{(n+1)} = \mu_m^{(n)} + \alpha \frac{o_m^{(n)}(x^{(n+1)})}{\pi_m^{(n)}} \left(x^{(n+1)} - \mu_m^{(n)} \right), \tag{19}$$

$$\begin{aligned} \Sigma_m^{(n+1)} = \Sigma_m^{(n)} + \alpha \frac{o_m^{(n)}(x^{(n+1)})}{\pi_m^{(n)}} \cdot \\ \cdot \left((x^{(n+1)} - \mu_m^{(n)})(x^{(n+1)} - \mu_m^{(n)})^T - \Sigma_m^{(n)} \right), \end{aligned} \tag{20}$$

where α determines the influence of the new sample on the old estimate. A component m is discarded when the weight θ_m becomes negative.

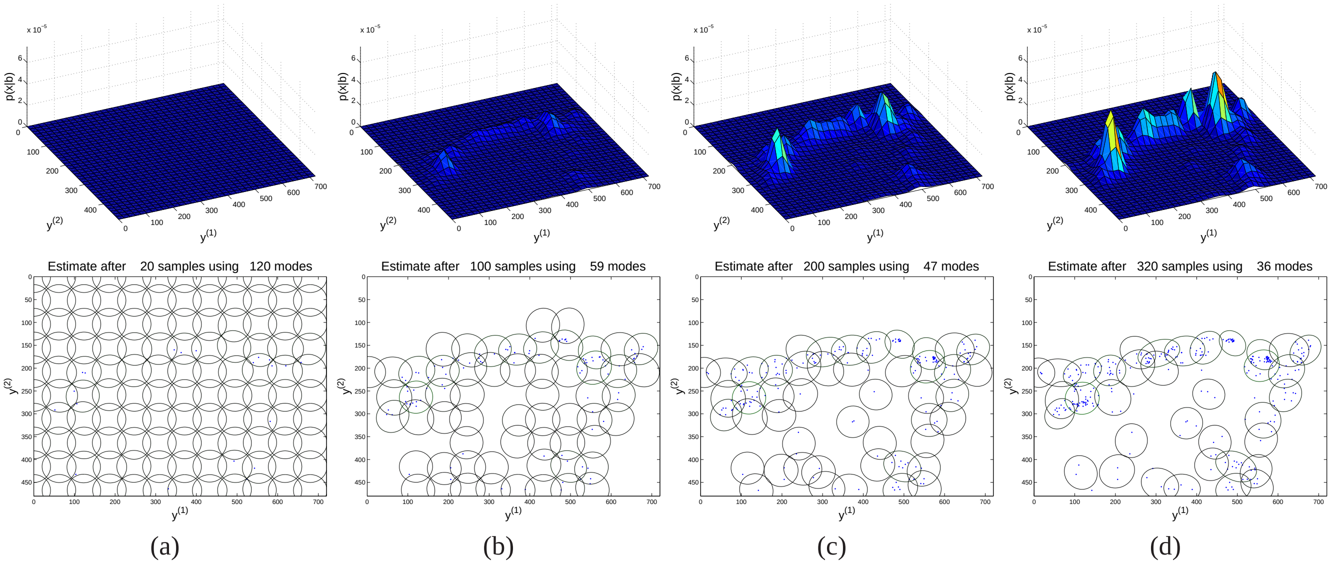
Although mixture-of-Gaussian components may produce a good approximation of the underlying distribution using a small number of parameters, the final result may not be appropriate for tracking. In fact, at initialization, when no prior information is available (i.e., $n = 0$), it is difficult to obtain a uniform distribution by mixing Gaussian components only. Either we initialize Σ_m with large values, or we distribute a large number of components on the data space. Both solutions result in a slower learning process. Moreover, after training, the probability of an event tends to zero far from the Gaussian center. If a birth or clutter event happens in these regions, then the tracking algorithm is likely to fail. A typical example of this problem is given by birth events generated by dynamic occlusions. In this case, to avoid a lost track, after an extended validation delay a rebirth should still be possible.

To overcome this problem, we use a non-homogeneous mixture composed of a uniform component, $u(x)$, and the GMM of Eq. (14). We approximate $p(x|b)$ with

$$p(x|b) \approx \pi_u u(x) + \pi_g p(x|\theta), \tag{21}$$

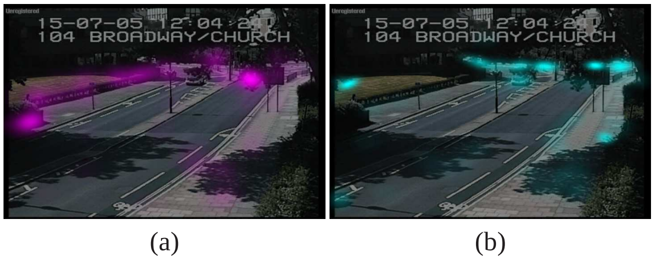
where $u(x) = \frac{1}{V} rect(x)$, V is the volume of the space, and π_u and π_g are the weights associated with the uniform component and with the Gaussian mixture. We set at initialization $\pi_u = 1$ and $\pi_g = 0$ so that we have an uninformative initial estimate. We also set $\pi_u = 10^{-3}$ as the minimum value that π_u can get during learning. Given this constraint, the algorithm refines $p(x|b)$ in a hierarchical fashion: we first use ML to compute π_u and π_g (i.e., Eq. (17) and (18) with $\tau = 0$), and we then update θ independently from π_u according to Eqs. (17)-(20). This approach introduces a bias in the estimate of the weights as the ownerships of Eq. (17) are computed using π_m and not $\pi_m \times \pi_g$. However, the update step of π_m and Σ_m does not depend on π_m (i.e., π_m simplifies by substituting Eq. (17) into Eq. (19) and Eq. (20)), and in practice with localized distributions and large n $\pi_u << \pi_g$ thus the bias tends to reduce with the amount of data available.

To learn the birth intensity from X_b we initialize a grid of 12x10 6D Gaussians equally spaced in the 2D positional state space and centered on zero speed and on the objects average size. The choice on the number of Gaussians depends on the complexity of the scene. However, as the components



(a) (b) (c) (d)

Fig. 7. Example of learning with recursive update of the birth density $p(x|b)$ for S101. The birth events used for density estimation are displayed in Fig. 6 (a). Although $p(x|b)$ is defined on a 6D state space, for visualization purpose we show the information related to the 2D position subspace only $(y^{(1)}, y^{(2)})$. First row: evolution of the birth density model with the number of samples processed. Second row: corresponding evolution of the GMM components.



(a) (b)

Fig. 8. Learned intensities for scenario S101 from the CLEAR-2007 dataset superimposed on the original images. (a) Birth intensity (note that the major modes are associated with entry areas). (b) Clutter intensities (note that waving vegetation produces clutter that is correctly modeled by the GMM).

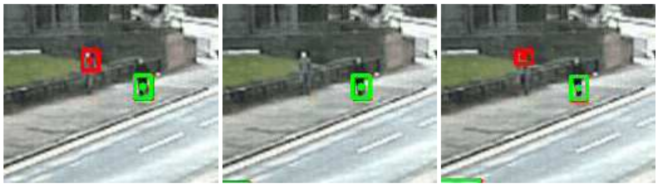


Fig. 9. Example of inconsistent detections interpreted by a PHD filter as clutter and therefore removed. Simple heuristics cannot differentiate these data from real clutter (red: observations; green: output of the PHD filter).

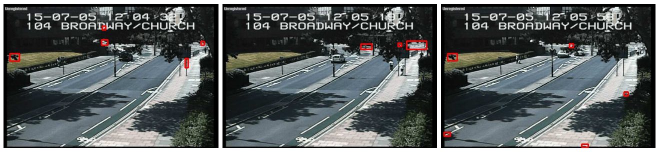
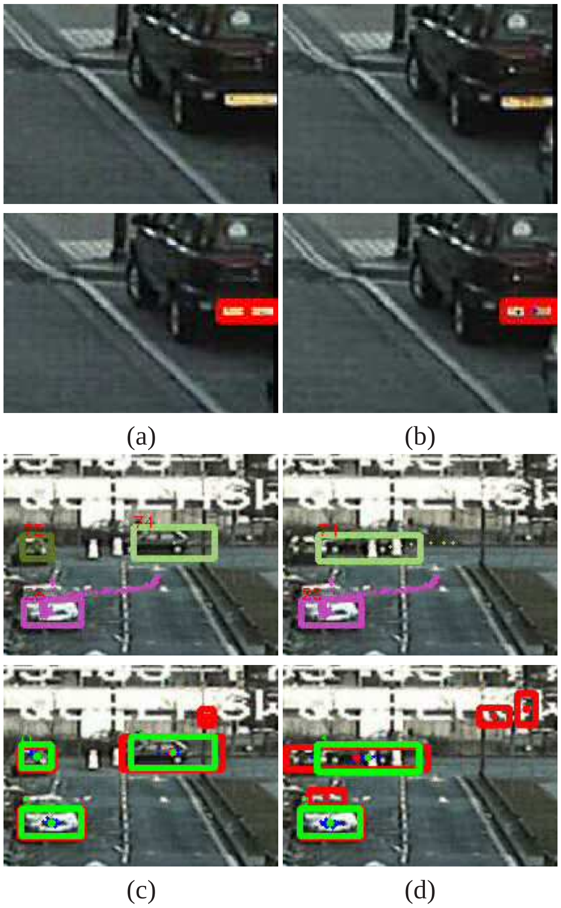


Fig. 10. Sample detections that are marked interactively as clutter and then used for density estimation.

are selected by the Dirichlet prior (Eq. (16)), we only need to overestimate the number of entry regions. Fig. 7 shows the evolution of the $p(x|b)$ estimate as more and more data become available. The Dirichlet prior reduces the weight of the modes that are not supported by sufficient evidence. After processing 320 trajectories (Fig. 7 (d) and Fig. 8 (a)) a few peaks (i.e., entry regions) are clearly visible. Two major peaks correspond to areas over the road where vehicles appear. Smaller peaks are visible on the sidewalks. The remaining components of the mixture model birth events caused by track re-initializations.

The procedure for the estimation of the clutter intensity $p_k(z|c)$ is similar to that of the birth intensity. However, the collection of the detections Z_c due to clutter is not performed automatically from the tracker output as these detection may contain the same errors we want to correct. Likewise for short trajectories as they could be generated either by tracking errors or by partially undetected real objects. Fig. 9 shows an example of a flickering detection on a small target with limited contrast with respect to the background.

Clutter data are therefore collected with an interactive procedure, which requires a minimal user intervention. After the detector is applied on a training set of frames, the user selects the detections that are not associated with objects of interest in randomly chosen frames. Fig. 10 shows sample detections selected as clutter and Fig. 6 (b) displays the centroids of the clutter data collected on a real-world surveillance scenario. Note that most of the clutter is in this case associated with waving vegetation and that a few false detections are also associated with high contrasted regions due to shadows. Given the sets of events $Z_c = \{z_{c,i}\}_{i=1}^{M_c}$, representing the locations and sizes of the cluttered observations, we learn the clutter intensity by initializing a grid of 16x14 4D Gaussians equally spaced in the 2D positional state space and centered on the objects average size. Because clutter data can be concentrated around small volumes of the observation space, we use a larger number of Gaussians than in the birth case to allow for higher spatial resolution. Fig. 11 and Fig. 8 (b) show an example of clutter density learned using 1800 false detections collected with user interaction on the results from scenario S101 (CLEAR-2007 dataset). The peaks of the probability


(a) (b)

(c) (d)

Fig. 12. Filtering results of the Particle-PHD filter using learned clutter and birth intensities (GM) on the same data as Fig. 4. First and third row: tracker output. Second and fourth row: the detections from a background subtraction algorithm are color-coded in red and the PHD output is color-coded in green. The filtering strength is modulated by the Gaussian Mixture birth and clutter models. (a)-(b): strong filtering in a background area. (c)-(d): weak filtering near an entry zone.

distributions (in violet) correspond to areas of the image where waving vegetation generates a large number of false detections.

The comparison between clutter and birth intensities in Fig. 8 (a)-(b) shows that in some cases regions with high birth rates overlap with regions with high clutter rates. This overlap might be only spatial, as the intensities also depend on the target size and, for birth intensities, on the initial target speed. However, if the detections associated to clutter and birth events have similar positions and sizes, then the balance between strong and weak filtering will be naturally determined by the system based on the statistics of the training data.

V. Experimental results

A. Experimental set-up

THIS section demonstrates the proposed multi-target tracking framework with learned intensities and assesses the contribution of learning clutter and birth density with mixture models. The *detector* used is a statistical change detector [23], followed by morphological filtering and connected component analysis. To facilitate experiments reproducibility the files containing the detector output Z_k are available at http://www.elec.qmul.ac.uk/staffinfo/andrea/PHD-MT.html.

The tests are conducted on two real-world urban *scenarios* from the CLEAR-2007 dataset (i.e., scenario S101 and S201). The videos have a frame size of 720×480 pixels and are recorded at 25Hz. The sequences contain global variations of illumination, light flickering and waving trees.

The objective *performance evaluation* follows the VACE-CLEAR protocol [31], which uses four scores, namely Multiple Object Detection Accuracy (MODA), Multiple Object Detection Precision (MODP), Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP). The detection scores are

$$\text{MODP}(k) = \frac{O_r}{N_m^{(k)}}, \tag{22}$$

where $N_m^{(k)}$ is the number of ground-truth detections mapped onto the tracker output, $O_r = \sum_{i=1}^{N_m^{(k)}} \frac{|G_i^{(k)} \cap D_i^{(k)}|}{|G_i^{(k)} \cup D_i^{(k)}|}$ quantifies the overlap between the i-th ground-truth object box $G_i^{(k)}$ and the mapped output detection $D_i^{(k)}$ in each frame k, and

$$\text{MODA}(k) = 1 - \frac{c_m(m_k^{(d)}) + c_f(fp_k^{(d)})}{N_G^{(k)}}, \tag{23}$$

where $c_m(.)$ and $c_f(.)$ are the cost functions[1] for the number of missing detections $m_k^{(d)}$ and false positives $fp_k^{(d)}$. Finally $N_G^{(k)}$ is the number of objects in the ground-truth at frame k. MODP and MODA are averaged over the number of frames of the evaluation segment, N_{fr}. The tracking scores are

$$\text{MOTP} = \frac{\sum_{i=1}^{N_m} \sum_{k=1}^{N_{fr}} \frac{|G_i^{(k)} \cap D_i^{(k)}|}{|G_i^{(k)} \cup D_i^{(k)}|}}{\sum_{j=1}^{N_{fr}} N_m^j} \tag{24}$$

and

$$\text{MOTA} = 1 - \frac{\sum_{j=1}^{N_{fr}} (c_m(m_j^{(k)}) + cs_f(fp_j^{(k)}) + log_e(id_{sw}))}{\sum_{i=1}^{N_{fr}} N_G^i}, \tag{25}$$

where N_m is the number of mapped objects over the entire track, $m_j^{(k)}$ is the number of missing tracks, $fp_j^{(k)}$ is the number of false positive tracks at frame j and id_{sw} is the number of false identity switches.

B. Discussion

To assess the impact of the learning on the tracking performance we compare the results of the proposed method (GM) using learned birth and clutter intensities against the results of the baseline tracker (UM) using a preset uniform distribution of birth and clutter. We trained the models on different frame spans than those used for the testing. After applying the tracker on the training frame spans using uniform clutter and birth intensities and the same set of manually tuned parameters as in our prior work [21], we extract birth and clutter samples from the tracker output (see Section IV) and use these samples to estimate birth and clutter intensity models. Finally, the trackers are tested on the evaluation segments where ground-truth data is available. The intensity magnitudes of UM ($\bar{r}$ and

[1] These functions are internally defined in the CLEAR evaluation toolbox, available at http://www.clear-evaluation.org/ (last accessed: November 2007).

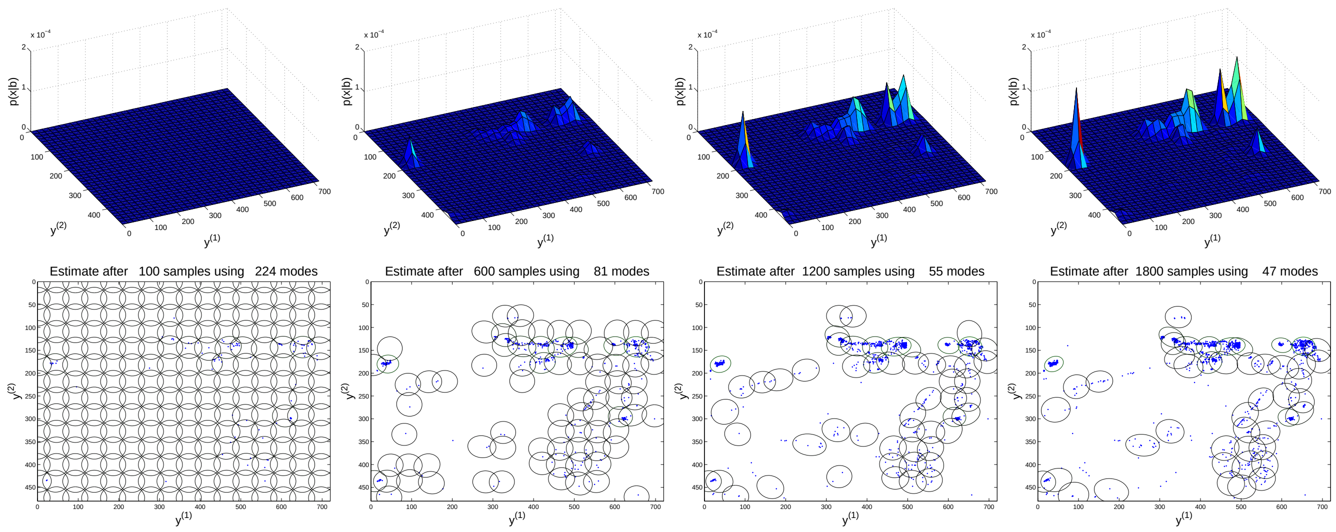


Fig. 11. Example of learning with recursive update of the clutter density $p(x|c)$ for S101. The input clutter events used are displayed in Fig. 6 (b). Although $p(x|c)$ is defined on a 4D observation space, for visualization purpose we show the information related only to the 2D position subspace only $(y^{(1)}, y^{(2)})$. First row: evolution of the clutter density model with the number of samples processed. Second row: corresponding evolution of the GMM components.

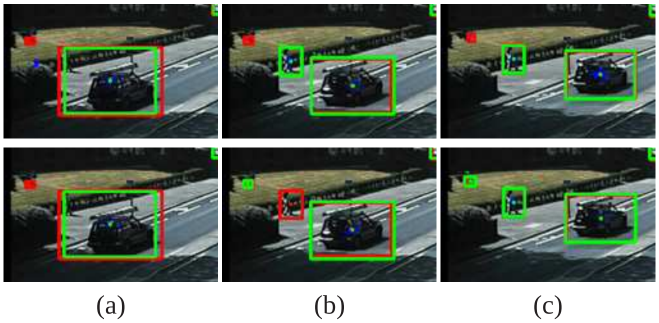


Fig. 13. Comparison of filtering results on the scenario S101. First row: tracker that uses learned clutter and birth intensities (GM). Second row: tracker that uses uniform intensities (UM). False detections due to waving trees are more consistently removed by using the Gaussian-Mixture-based birth and clutter models (red: detections; green: PHD filter output).

$\bar{s}$) are the same as in the learning phase. We also compare these two solutions with six other algorithms obtained by combining different clutter and birth learning strategies: GMM birth and uniform clutter intensities (A1); uniform birth and GMM clutter (A2); uniform birth and clutter but magnitudes $\bar{r}$ and $\bar{s}$ estimated from the data (A3); clutter as in A3 and GMM birth (A4); birth as in A3 and GMM clutter (A5); GMM birth and clutter intensities, but with birth interactive data collection, performed as for the clutter data (A6).

Figure 12 shows sample results of GM on scenario S201 where contextual feedback improves the PHD filter performance. As low birth intensity (i.e., strong temporal filtering) is estimated over the parking areas (Fig. 12 (a)-(b)), false detections on the number plate are consistently removed (Fig. 12 (a)-(b)). Compare these results with those of UM in Fig. 4. On the same scenario high birth intensity (i.e., weak filtering effect) is applied to the entry regions. This allows for correct detection and tracking of a fast car in the camera far-field (Fig. 12 (c)-(d)). Similar considerations are valid for the clutter model. When clutter is localized, the GMM-based density estimation introduces further degrees of freedom for filter tuning. Fig. 13 shows a comparison of the GM and UM filtering results on scenario S201. The detections corresponding to waving branches are filtered out for a longer number of frames due to the feedback from the GMM clutter model (Fig. 13 (b)-(c)). Low clutter levels instead are assigned to the sidewalk regions, thus allowing the PHD filter to validate after few frames the coherent detections corresponding to a pedestrian (Fig. 13 (a)-(b)).

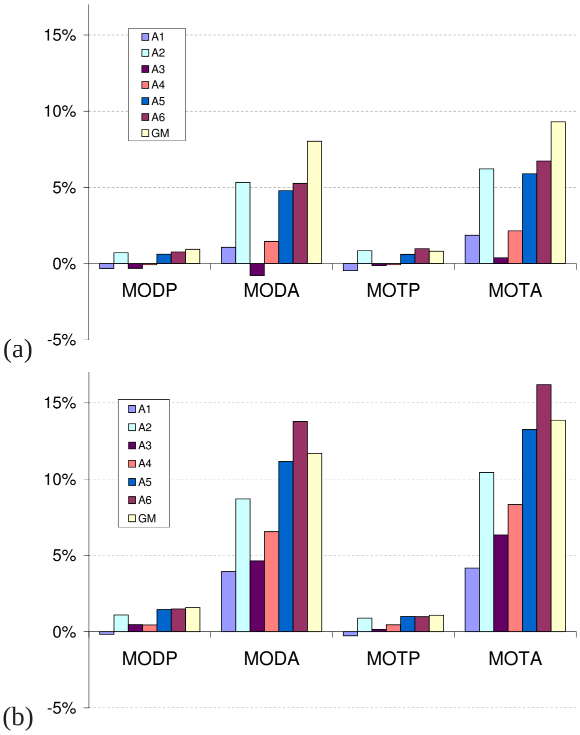


Fig. 14. Comparison of tracking results on the CLEAR-2007 scenarios S101 (a) and S201 (b). The bars represent the score percent difference with respect to the base-line algorithm (UM). The series A1-A6 were obtained with different context learning strategies. See the text for details.

Figure 14 compares the tracking results on the two scenarios from the CLEAR-2007 dataset. The values of the bars are the percent score differences with respect to the base-line tracker UM. In both scenarios the Gaussian mixtures used to model birth and clutter intensities (GM and A6) outperforms the other models, especially in terms of accuracy. GM and A6 improve the clutter removal capabilities of the PHD filter, thus reducing false detections and false tracks. This is also confirmed by the results in Fig. 15 obtained by varying the values of clutter and birth magnitudes $\bar{r}$ and $\bar{s}$ in UM. In all cases GM outperforms UM in terms of accuracy. This is true also for the precision scores except when the clutter intensity is overestimated. However, in this case UM achieves slightly better precision than GM, but at the cost of a large drop of accuracy (Fig. 15 (d)). It is important to note that the curves produced by UM are stable around their maximum values as by changing $\bar{r}$ and $\bar{s}$ the filtering behavior becomes more suitable on a subset of targets but sub-optimal on another. This leads to similar performance scores. Also, the results in Fig. 14 show that, given the same average intensity, the GMM density estimates improve the performance with respect to the uniform distributions (compare GM with A3). Both clutter and birth intensity models contribute to the final performance improvement. However, clutter intensity trained with manually labeled data achieves better results than GM birth intensity trained using the output of the tracker (compare A1 with A2, or A3 with A4). This is due to the fact that the birth model must account also for track re-initializations; the volume of the state space where a birth event is likely to happen is larger and thus the model is less discriminative than that for clutter. However, a more precise birth model trained with manually annotated data (A6) leads to ambiguous results (compare A6 and GM). On the one hand, when most false detections are generated by background clutter, as in scenario S201 (Fig. 14 (b)), a tighter birth constraint allows A6 to outperform GM in terms of accuracy. On the other hand, when a large percentage of tracking errors is due to occlusions and blob merging (as in scenario S101), the same constraint prevents a prompt reinitialization of the tracks (Fig. 14 (a)).

We compared the performance of the proposed algorithm (PROP) with that of the data association (DA) method from [8] and the multiple hypothesis tracker (MHT) [6] on the same detections for all trackers. Also, as due to the Kalman propagation of the hypotheses in the MHT implementation we could not apply the SDV models used by the PHD filter, for a fair comparison we resorted to two models with same transition and observation matrices, but fixed variances. The parameters of MHT were manually set by visually inspecting the final tracking result and we report them here for reproducibility of the results. The variances for position measurement noise was 3 pixels; for the size measurement noise: 6 pixels; for the velocity state noise: 1.5; for the size state noise: 4 pixels; for the initial velocity state: 200. The detection probability is 0.97; $\lambda_x = 150$; the mean new targets is set to 0.0025; the mean false alarms to 0.00007; the maximal Mahalanobis distance is 20; the tree depth is 10; the minimum likelihood ratio is 0.001; the maximum number of hypotheses is 300.

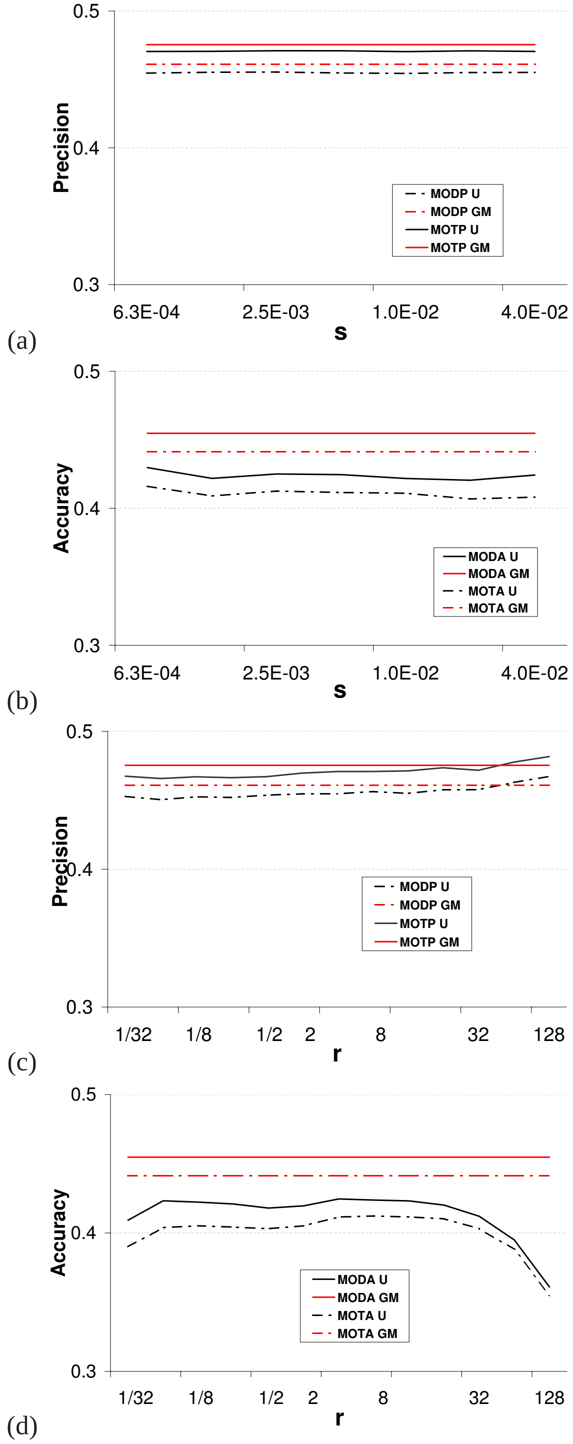


Fig. 15. Comparison of tracking performance on Scenario S101 when varying the birth and clutter magnitudes ($\bar{r}$ and $\bar{s}$) between the tracker with Gaussian-Mixture-based birth and clutter intensities (GM) and the tracker with uniform distributions (UM).

Figure 16 shows that the proposed approach outperforms the other two trackers in terms of accuracy. In particular, the lack of an explicit clutter model heavily affects the performance of DA. In fact, DA validates detections using a simple procedure based on the distance from the prediction. MHT copes better with the challenges of real-world tracking scenarios due to a better clutter model. Nevertheless, the PHD outperforms MHT in terms of accuracy in both scenarios and has similar

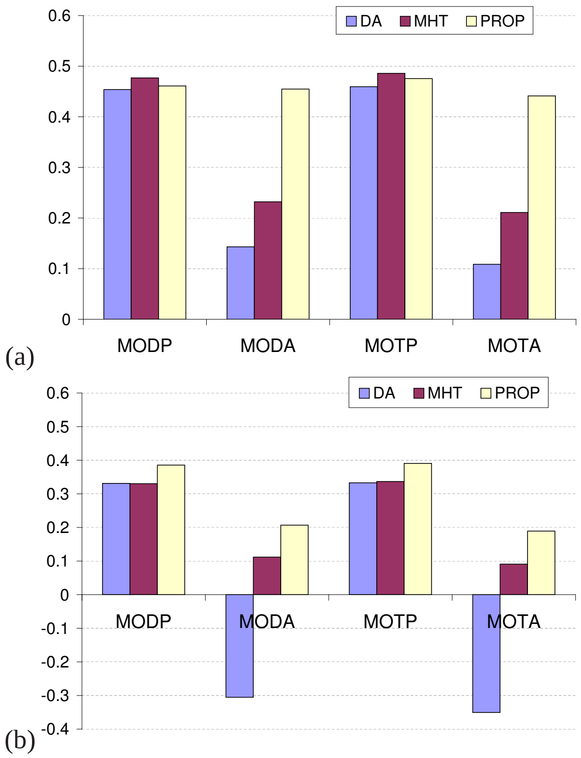


Fig. 16. Comparison of tracking performance between the proposed methodology (PROP), the data association method from [8] (DA) and Reid's multiple hypothesis tracker (MHT). (a): Scenario S101. (b): Scenario S201.

precision in scenario S101 (Fig. 16 (a)) and better precision in the more challenging scenario S201 (Fig. 16 (b)). Three factors contribute to the improved performance: first, unlike MHT, the PHD filter does not explicitly postulate association hypotheses, and this is advantageous with ambiguous associations. Second, the Montecarlo approximation used by the PHD is more flexible in terms of model choice than the Kalman filter. Third, the ability to model contextual information enhances the performance of the PHD filter.

Finally, note that for simplicity, but without loss of generality, in the models we assumed stationarity for clutter and birth events. However, when the scene undergoes significant illumination changes, a non-stationary clutter model may be necessary. For example, Fig. 5 (b) shows that cluttered detections may be localized close to the borders of strong shadows, whose position is time variant. Similar considerations can be made for the changes of birth intensity caused by different traffic levels. In both cases, a jump Markov system [32] could be used to switch between multiple models based on external triggers (e.g., time constraints) or on content-based cues (e.g., results of the scene analysis) [33].

VI. Conclusions

We presented a framework to learn contextual information from tracking data and user feedback. The results of the learning enable a multi-target Bayes filter to spatially adapt its behavior. A parametric model based on Gaussian mixture is used to estimate state and observation dependent birth and clutter intensities. These models are used as input for a multi-target tracker based on the PHD filter to adapt its response according to the position in the target state space.

Experimental results on real-world data show that it is possible to learn contextual information via a combination of automated and interactive feedback from the tracker, and that the proposed framework improves the capability of the PHD filter in removing persistent clutter, and reduces the filter delay in regions where the birth event is likely to happen according to the learned model. The performance improvement is due to the space-dependent birth and clutter models. The clutter model strengthens the filter in presence of spatially localized clutter and weakens the filter in clutter-free regions. The birth model instead allows us to increase the filtering strength where targets are unlikely to appear. When compared with uniform birth and clutter models the combined space-dependent models (i) reduce the detection latency of the recursive filter in clutter-free areas and (ii) reduce the number of false tracks generated by persistent clutter. The proposed approach is general and can be applied to any multi-target Bayes tracker capable of position-dependent birth and clutter modeling.

Future work includes the extension of the proposed framework to continuous learning, and to non-stationary clutter and birth models. In the case of continuous learning, additional data could be fed to the recursive GMM to update on-line clutter and birth intensities and a study is necessary to guarantee the convergence of the GMM algorithms to a meaningful solution. Also, the continuous learning approach should have the capability to recognize wrong models to be removed and recomputed. Ideally, the design of non-stationary clutter and birth models should cope with both the temporal evolution of a static context and with the movements of a camera. In the latter case the learning approach should condition birth and clutter intensities not to the absolute spatial location in the image but to the relative location with respect to recognizable landmark objects like roads, doors and vegetation. Finally, although the PHD filter can cope with sporadic missing detections, a relevant source of error is associated with inter-object occlusions (i.e., blob merging and splitting). In these conditions classifier–based trackers [34] could improve the results of change–detection–based trackers.

References

[1] B. Bose and E. Grimson, "Learning to use scene context for object classification in surveillance," in *Joint IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, Nice, France, Oct. 2003, pp. 94–101.

[2] H. Nguyen, Q. Ji, and A. Smeulders, "Spatio-temporal context for robust multitarget tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 29, no. 1, pp. 52–64, Jan. 2007.

[3] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[4] E. Maggio and A. Cavallaro, "Hybrid particle filter and mean shift tracker with adaptive transition model," in *Proc. of IEEE International Conf. on Acoustics, Speech, and Signal Processing*, vol. 2, Philadelphia, USA, Mar. 2005, pp. 221–224.

[5] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automat. Contr.*, vol. 24, no. 6, pp. 843–854, 1979.

[6] I. J. Cox and S. L. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 2, pp. 138–150, 1996.

[7] J. Vermaak, S. Godsill, and P. Perez, “Monte Carlo filtering for multi-target tracking and data association,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 1, pp. 309–332, 2005.

[8] K. Shafique and M. Shah, “A noniterative greedy algorithm for multi-frame point correspondence,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 1, pp. 51–65, 2005.

[9] B. Song and A. Roy-Chowdhury, “Stochastic adaptive tracking in a camera network,” *Proc. of International Conf. on Computer Vision*, pp. 1–8, Oct. 2007.

[10] A. Doucet, B. Vo, C. Andrieu, and M. Davy, “Particle filtering for multi-target tracking and sensor management,” in *Proc. of International Conf. on Information Fusion*, vol. 1, 2002, pp. 474–481.

[11] Y. Boers and J. Driessen, “Multitarget particle filter track before detect application,” *Radar, Sonar and Navigation, IEE Proceedings*, vol. 151, no. 6, pp. 351–357, 2004.

[12] S. Sarkka, A. Vehtari, and J. Lampinen, “Rao-Blackwellized particle filter for multiple target tracking,” *Information Fusion*, vol. 8, no. 1, pp. 2–15, Jan. 2007.

[13] Q. Yu, G. Medioni, and I. Cohen, “Multiple target tracking using spatio-temporal markov chain monte carlo data association,” *Proc. of IEEE Conf. on Comp. Vis. and Pattern Recog.*, pp. 1–8, June 2007.

[14] R. Mahler, “A theoretical foundation for the Stein-Winter Probability Hypothesis Density (PHD) multitarget tracking approach,” in *Proc. 2002 MSS Nat’l Symp. on Sensor and Data Fusion*, vol. 1, San Antonio, USA, Jun. 2000.

[15] B. Vo, S. Singh, and A. Doucet, “Sequential Monte Carlo implementation of the PHD filter for multi-target tracking,” in *Proc. of International Conf. on Information Fusion*, vol. 2, Cairns, AU, Jul. 2003, pp. 792–799.

[16] H. Sidenbladh and S. Wirkander, “Tracking random sets of vehicles in terrain,” in *Proc. of IEEE Workshop on Multi-Object Tracking*, Madison, USA, Jun. 2003.

[17] D. Clark, I. Ruiz, Y. Petillot, and J. Bell, “Particle PHD filter multiple target tracking in sonar images,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 1, pp. 409–416, 2006.

[18] N. Ikoma, T. Uchino, and H. Maeda, “Tracking of feature points in image sequence by SMC implementation of PHD filter,” in *Proc. of SICE Annual Conf.*, vol. 2, Sapporo, JP, Aug. 2004, pp. 1696–1701.

[19] Y. Wang, J. Wu, A. Kassim, and W. Huang, “Tracking a variable number of human groups in video using probability hypothesis density,” in *Proc. of IEEE International Conf. on Pattern Recognition*, vol. 3, Hong Kong, CH, Aug. 2006, pp. 1127–1130.

[20] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro, “Particle PHD filtering for multi-target visual tracking,” in *Proc. of IEEE International Conf. on Acoustics, Speech, and Signal Processing*, vol. 1, 2007, pp. I–1101–I–1104.

[21] E. Maggio, M. Taj, and A. Cavallaro, “Efficient multi-target visual tracking using random finite sets,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1016–1027, Aug. 2008.

[22] Z. Zivkovic and F. Van der Heijden, “Recursive unsupervised learning of finite mixture models,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 5, pp. 651–656, 2004.

[23] A. Cavallaro and T. Ebrahimi, “Interaction between high-level and low-level image analysis for semantic video object extraction,” *EURASIP Journal on Applied Signal Processing*, vol. 6, pp. 786–797, Jun. 2004.

[24] T. E. D. Makris, “Automatic learning of an activity-based semantic scene model,” in *Proc. of IEEE International Conf. on Advanced Video and Signal Based Surveillance*, Miami, USA, Jul. 2003, pp. 183–188.

[25] X. Wang, K. Tieu, and E. Grimson, “Learning semantic scene models by trajectory analysis.” in *Proc. of the European Conf. on Computer Vision*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3953. Springer, 2006, pp. 110–123. [Online]. Available: http://dblp.uni-trier.de/db/conf/eccv/eccv2006-3.html#WangTG06

[26] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, Eds. CRC Press, 1988, vol. 37, no. 1.

[27] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[28] M. Figueiredo and A. K. Jain, “Unsupervised learning of finite mixture models,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 3, pp. 381–396, 2002.

[29] M. Brand, “Structure learning in conditional probability models via an entropic prior and parameter extinction.” *Neural Computation*, vol. 11, no. 5, pp. 1155–1182, 1999.

[30] D. M. Titterington, “Recursive parameter estimation using incomplete data,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 46, pp. 257–267, 1984.

[31] R. Kasturi, *Performance evaluation protocol for face, person and vehicle detection & tracking in video analysis and content extraction*, Computer Science & Engineering University of South Florida, Jan. 2006.

[32] A. Pasha, B. Vo, H. Tuan, and W.-K. Ma, “Closed form phd filtering for linear jump markov models,” *Proc. of International Conf. on Information Fusion*, pp. 1–8, July 2006.

[33] F. Bremond and M. Thonnat, “A context representation for surveillance systems,” in *ECCV Worshop on Conceptual Descriptions from Images*, April 1996.

[34] T. Zhao and R. Nevatia, “Tracking multiple humans in complex situations,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 9, pp. 1208–1221, 2004.

Emilio Maggio received the M.Sc. degree in telecommunication engineering from the University of Siena, Italy in 2003 and the Ph.D. degree in Electronic Engineering from Queen Mary, University of London, UK in 2008. He is now Computer Vision Scientist at VICON, the motion capture company part of the Oxford Metric Group. In 2007 he visited the Mitsubishi Research Labs (MERL), Cambridge US, to work on video object tracking. Also, in 2003 he visited the Signal Processing Institute at the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, working in the area of video coding. His research interests are object tracking, classification, Bayesian filtering, sparse image and video coding. In 2005 and 2007 he was awarded twice a best student paper prize at ICASSP, the IEEE International Conference on Acoustics, Speech, and Signal Processing. He was also a member of the Team, winner of CSIDC 2002, the IEEE Computer Society 3rd Annual International Design Competition. He serves as a reviewer for IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Signal Processing, Signal Processing (Elsevier), ACM Multimedia, and a number of international conferences.

Andrea Cavallaro received the M.Sc. degree (summa cum laude) from the University of Trieste, Italy, in 1996 and the Ph.D. degree from the Swiss Federal Institute of Technology (EPFL), Lausanne, in 2002, both in electrical engineering. From 1998 to 2003, he was a Research Assistant with the Signal Processing Laboratory, EPFL. Since 2003, he has been with Queen Mary University of London, U.K., where he is Reader in Multimedia Signal Processing (Associate Professor). Dr. Cavallaro was the recipient of a Research Fellowship with BT, the Royal Academy of Engineering Teaching Prize in 2007, and was coauthor for two student paper prizes at IEEE ICASSP 2005 and 2007. He is an Associate Editor for the IEEE Signal Processing Magazine, the IEEE Transactions on Multimedia, and the IEEE Transactions on Signal Processing. He is an elected member of the IEEE Signal Processing Society, Multimedia Signal Processing Technical Committee, Guest Editor of the Special Issues on ’Multi-sensor object detection and tracking’, Signal, Image and Video Processing Journal (Springer); on ’Video Tracking in Complex Scenes for Surveillance Applications’, Journal of Image and Video Processing, on Multi-camera and multi-modal sensor fusion, Computer Vision and Image Understanding (Elsevier), and on Video Analytics for Surveillance, IEEE Signal Processing Magazine. Dr. Cavallaro serves as General Chair for IEEE/ACM ICDSC 2009 and BMVC 2009; has served as General Chair for M2SFA2 2008, SSPE 2007, and IEEE AVSS 2007, and Technical Program chair of the European Signal Processing Conference (EUSIPCO 2008). He has acted as expert evaluator for the National Research Agency (ANR), France; the European Commission; the EPSRC, UK; Microsoft Research, UK; the National Science Foundation, USA; and the National Science Foundation, Switzerland. He is a steering committee member for IEEE AVSS and has been a member of the organizing/technical committee of several conferences, including IEEE ICME, IEEE ICIP, SPIE VCIP, ACM Multimedia, IEEE AVSS, ACM/IEEE ICDSC, ECCV-VS, PETS. Dr. Cavallaro has edited with H. Aghajan the book Multi-Camera Networks: Principles and Applications (Elsevier) and has authored more than 90 papers, including 6 book chapters.