This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

DRDC-RDDC-2017-P029

The Social Force PHD Filter for Tracking Pedestrians

K. Krishanth, X. Chen, R. Tharmarasa and T. Kirubarajan
ECE Department, McMaster University
Hamilton, Ontario, Canada
Email: {krishk8, chenx73, tharman, kiruba}@mcmaster.ca
Mike McDonald
Defence Research and Development Canada
Ottawa, Ontario, Canada
Email: mike.mcdonald@drdc-rddc.gc.ca

***Abstract*—This paper addresses the problem of tracking multiple pedestrians whose motion is dependent on one another. The behavior of a pedestrian may be often affected by the motion of other pedestrians, obstacles in the surrounding and his/her intended destination. Hence, a motion modeling technique, which integrates the various factors that affect the motion of pedestrians, is needed. In this paper, a social force based motion model integrated into the Probability Hypothesis Density (PHD) framework is proposed. The social force concept has previously been used to model pedestrian motion when there are interactions among pedestrians. In this paper, the Sequential Monte Carlo (SMC) technique and the Gaussian Mixture (GM) technique are used to implement the proposed Social Force PHD (SF-PHD) filter and its multiple model variant in pedestrian tracking scenarios. A particle labeling approach is used in the SMC technique while a Gaussian component labeling approach is used in the Gaussian mixture technique for this purpose. Also, a modified performance measure independent of the proposed approaches but based on the Posterior Cramer-Rao Lower Bound (PCRLB) for targets whose motion is dependent on one another is derived. Simulation and real data-based results show that both the SMC implementation and the Gaussian mixture implementation of the proposed SF-PHD filter outperform existing filters that assume independent motion among ground targets.**

Keywords: pedestrian tracking, social force PHD filter, dependent ground target motion, error bounds for dependent targets, ground target tracking, urban tracking.

I. INTRODUCTION

Ground target tracking has applications in surveillance, ground traffic control, pedestrian safety and perimeter surveillance [17][30][35]. To track ground targets well, motion models that accurately describe their motion are needed. Current approaches used in the ground target tracking literature often assume that the motion of a target is independent of the other targets and of the environment in which they evolve [15][35]. However, the actual motion of a ground target (e.g., pedestrian) might be complex and influenced by environmental factors and other targets [16][25][28]. This necessitates a sophisticated motion model for ground target tracking, specifically pedestrian tracking. This issue has been addressed in pedestrian motion modeling literature, where one pedestrian's motion is dependent on both the motion of other pedestrians and environmental factors [13][16][24][32].

The challenge in accurately modeling the pedestrian motion arises from the coordinated or inter-related movement along lanes or in clusters due to interactions and dependencies among individuals [2]. This has led to two different modeling techniques: microscopic and macroscopic [14][32]. In the microscopic approach, pedestrians are considered as a group of individuals whose inter-related movements arise due to complicated interactions among group members and with the environment [2]. In the macroscopic approach, pedestrians are considered as a flow and interactions among pedestrians are not modeled explicitly. Therefore, the macroscopic approach is not suitable to track a pedestrian's motion individually or in buildings with obstacles or in pedestrian zones [32]. Thus, the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

focus of this paper is on leveraging microscopic motion models for pedestrian tracking. In [10], macroscopic cell-transmission model is applied to estimate the traffic model parameters in the presence of clutter. Estimating the traffic model parameters using a gas-kinetic approach to traffic-flow theory is proposed in [20]. Similarly, the macroscopic models can be used to extract certain relevant information of pedestrian flows [11] (e.g., entrance and exit of a building, average speed of a pedestrian group), which can then be used as prior information for microscopic models.

There are several microscopic pedestrian simulation models in the literature [14][32]. For example, the Benefit Cost Cellular Model [12] simulates a cell-based particle approach for each pedestrian. Each cell is given a score based on the proximity to the pedestrians and only one pedestrian can be assigned to a cell. The score reflects the gain of the pedestrian in his/her motion towards the destination. Similar to the Benefit Cost Cellular model, the Cellular Automaton (CA) model [6] simulates a cell-based entity approach. A pedestrian's path is represented in the model as grid cells. Cell hopping and lane changing of the pedestrian movements are naturally considered in the modeling. Both of these models are discrete in nature, and are not well suited for integration into a probabilistic tracker due to insufficient information on the error propagation mechanism.

In the Magnetic Force Model [22], it is assumed that pedestrian movement can be modeled by magnetic field equations as pedestrians move to their destination while avoiding collisions. Each pedestrian or obstacle is assumed to be positive pole and the destination of the pedestrian is assumed to be the negative pole. All forces from the obstacles, other pedestrians and the destination are modeled based on the magnetic force equations and the velocity of each pedestrian is updated based on the sum of the calculated forces. The Social Force Model [13] was developed based on the same idea as both the Magnetic Force Model and the Benefit Cost Cellular Model. In this model, it is assumed that a pedestrian's motion is affected by various social forces such as the motivational force from the goal and the repulsion forces from other pedestrians and obstacles. The summation of all the social forces provides an acceleration for the motion of the pedestrian. Among the existing pedestrian motion models, the social force model is the most amenable to stochastic modeling and integration into a probabilistic estimation framework. Thus, the social force model approach is used in this paper in the prediction step of the proposed Probability Hypothesis Density filter framework for tracking inter-related pedestrian motion.

A Sequential Monte Carlo Multitarget Bayes (SMC-MTB) filter based on Random Finite Sets (RFS) is used to track pedestrians in [28]. The social force model is used as an additional weighting factor for the predicted multitarget particles instead of modifying the directions. The Interacting Multiple Model (IMM) filter is a computationally efficient algorithm for tracking maneuvering targets [9] and the Multiple Hypothesis Tracking (MHT) algorithm [8] is a widely used approach for solving multitarget data association problems. The combined IMM-MHT tracker is developed in [7]. In [16], the acceleration component provided by the social force model is used as a zero-mean Gaussian noise and integrated into the IMM-MHT framework. The assumption of zero-mean Gaussian noise for social forces is not valid and the cross-covariance component between the targets is not considered in the covariance propagation in [16]. As an alternative to the IMM-MHT framework, the IMM technique combined with the Joint Integrated Probabilistic Data Association (IMM-JIPDA) algorithm can be used to integrate social forces via the acceleration component. These algorithms serve as the baseline for comparison with the proposed technique.

The Probability Hypothesis Density (PHD) filter is a suboptimal [18] but computationally manageable alternative to the random finite sets theory that provides a practical solution to multitarget tracking problems. It is a recursive mechanism to propagate the first-order statistical moment, which is the posterior expectation, instead of the full posterior distribution. The PHD filter can be specifically used in cases with an unknown target number and unknown multitarget states. In general, the PHD filter equations include high-dimensional integrals and offer no closed-form solution. In [37], the Gaussian Mixture PHD (GM-PHD) filter, which provides the recursions for propagating the means, covariances and weights of the constituent Gaussian components of the posterior intensity under certain linear Gaussian assumptions, is derived. In the implementation of the GM-PHD filter, the number of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

Gaussian components increases as time progresses. Hence, pruning and merging approaches are used to solve this implementation/computational issue, which results in the GM-PHD filter approximating the density function of the PHD filter. In non-linear non-Gaussian cases, the Sequential Monte Carlo (SMC) technique [36] is an efficient way to approximate the density function by recursively propagating a group of weighted particles.

In this paper, a social force based motion model integrated into the Probability Hypothesis Density (PHD) framework is proposed. Two implementations, the SMC and the Gaussian mixture ones, for the proposed Social Force PHD (SF-PHD) filter are provided. The standard PHD filter implementations [36][37] keep no record of track identities and the estimation of an individual target cannot be obtained directly. In the proposed SF-PHD filter approach, a target's predicted state at the next time step is dependent on all targets' state at the current time step. Hence, a target's identity has to be maintained throughout the filtering process. A particle labeling approach is used in the proposed Sequential Monte Carlo Social Force PHD (SMC-SF-PHD) implementation. Further, during the resampling step, it is possible that no particle becomes available for some closely-spaced targets (e.g., sensor measurement covariance is relatively high compared to the distances between the nearby targets). To overcome this problem, resampling and clustering with label [38] are used in this paper. In the Gaussian Mixture Social Force PHD (GM-SF-PHD) implementation, the mean and the covariance of a target has to be calculated. Since the motions of targets are dependent on one another, existing implementation from [37] cannot be used as it is. To factor the effect of target motion dependency, equations are derived for correlated covariance matrix and a stack-based approach is used in the proposed GM-SF-PHD filter. Further, a target's identity is maintained by labeling the Gaussian components and propagating the labels in the subsequent time steps [23].

The Posterior Cramer-Rao Lower Bound (PCRLB) is defined as the inverse of the Fisher Information Matrix, and provides an error bound on the optimal achievable accuracy in state estimation [34]. The standard single target PCRLB [34] was extended to the multitarget case in [33]. However, it is a bound derived based on the assumption that the motions of different targets are independent of each other. In this paper, an error bound independent of the proposed SMC-SF-PHD filter and the GM-SF-PHD filter but based on the multitarget PCRLB is derived for the targets whose motions are dependent on each other. The new bound reflects the true achievable accuracy in the presence of target motion uncertainty with inter-dependency.

The remainder of the paper is structured as follows. The Social force model is discussed in Section II and the proposed Social Force PHD filter is given in Section III. An error bound based on PCRLB for dependent targets is given in Section IV. Results with simulated data and real data using single-model and multiple-model SF-PHD filters are given in Section V and concluding remarks are provided in Section VI.

II. The Social Force model

The social force model was developed to describe the motions of pedestrians [13][14]. These social forces are not the physical forces felt directly or generated by a pedestrian. Rather, they are a measure of a pedestrian's personal desire or aversion to move in a certain manner. The social force model takes into consideration the effects of the motion of other pedestrians and the obstacles or destinations in the environment to describe the motion of a pedestrian [16]. The sum of all the applied forces influencing the motion of pedestrian ϕ^i at time step $k-1$, ϖ^i_{k-1}, is given as follows:

$$\begin{aligned}\varpi^i_{k-1}(\phi^i,\mho,O) &= \alpha^i_{k-1}(\phi^i)+\beta^i_{k-1}(\phi^i,\mho,O)\\ &\quad+\gamma^i_{k-1}(\phi^i,\mho,O)\end{aligned} \tag{1}$$

The main forces that dictate the motion of a pedestrian in (1) are defined as follows:

1. Forces from personal motivation:

The model assumes that at time step $k-1$ pedestrian ϕ^i with mass m^i moves with velocity v^i_{k-1} and intends to reach a certain goal. Therefore, pedestrian ϕ^i wants to adopt to a certain desired velocity $\bar{v}^i$ in a desired direction $\bar{e}^i_{k-1}$ within a so called relaxation time τ^i. This personal motivation can be represented by an acceleration component as follows:

$$\alpha^i_{k-1}(\phi^i)=\left(\frac{\bar{v}^i\bar{e}^i_{k-1}-v^i_{k-1}}{\tau^i}\right) \tag{2}$$

2. Forces from other pedestrians and obstacles

The pedestrian might not be in a situation to retain the desired direction and desired velocity due to the effects of other pedestrians and the obstacles in the surroundings. The repulsive forces from these interactions are given by a force β^i_{k-1}. This force stops

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

a pedestrian from colliding with other obstacles and pedestrians. It is represented as a summation of forces influenced by static obstacles, denoted by subscript o, or other pedestrians ϕ^j as follows:

$$\beta^i_{k-1}(\phi^i, \mho, O) = \sum_{j \in \mho \setminus \{i\}} \beta^i_{k-1}(\phi^i, \phi^j) + \sum_{o \in O \setminus \{i\}} \beta^i_{k-1}(\phi^i, \phi^o) \tag{3}$$

with the set of all pedestrians being represented by $\mho = \{\phi_i\}_{i=1}^N$ and the static objects in the surrounding being represented by O. The repulsion force decreses proportional to the distance from the corresponding sources and in this paper it is modeled as

$$\beta^i_{k-1}(\phi^i, \phi^\kappa) = \frac{p^{i,\kappa}_{k-1}}{m^i} a^\kappa e^{\frac{r^{i,\kappa} - d^{i,\kappa}_{k-1}}{b^\kappa}} n^{i,\kappa}_{k-1} \tag{4}$$

where $\phi^\kappa \in \mho \cup O$ is either a pedestrian or a static object in the surroundings, $p^{i,\kappa}_k$ is the weight of the force from other pedestrians and obstacles , a^κ specifies the magnitude parameter of the exerted force and b^κ is the range parameter of the exerted force. The Euclidean distances between an entity ϕ^κ and a pedestrian ϕ^i is calculated based on the assumption that the objects and pedestrians in the environment are circular in shape with radii r^κ and r^i, respectively. Then, distance $d^{i,\kappa}_{k-1}$ is calculated between the centers of the pedestrian and entity, and the sum of their radii is $r^{i,\kappa}$. The direction of the force is given by the normalized vector $n^{i,\kappa}$ pointing towards ϕ^i from ϕ^κ.

3. Forces from physical constraints

Apart from the personal motivation and complex behavior towards the other pedestrian and objects, physical constraints of the environment also influence the motion of a pedestrian. For example, a pedestrian may try to keep a distance from walls, streets, obstacles and edges of buildings. The force that corresponds to this behavior is represented by γ^i_{k-1} onto pedestrian ϕ^i given as

$$\gamma^i_{k-1}(\phi^i, \mho, O) = \sum_{j \in \mho \setminus \{i\}} \gamma^i_{k-1}(\phi^i, \phi^j) + \sum_{o \in O \setminus \{i\}} \gamma^i_{k-1}(\phi^i, \phi^o) \tag{5}$$

$$\gamma^i_{k-1}(\phi^i, \phi^\kappa) = \frac{p^{i,\kappa}_{k-1}}{m^i} c^\kappa g(r^{i,\kappa} - d^{i,\kappa}_{k-1}) n^{i,\kappa}_{k-1} \tag{6}$$

where c^κ is the magnitude parameter of the force and g is the function to model the force from physical constraint.

The sum of the applied forces is given by ϖ^i_{k-1} in (1) and the basic equation for the change of pedestrian motion over time can be written as

$$\frac{d}{dt} v^i = \varpi^i_{k-1}(\phi^i, \mho, O) \tag{7}$$

A. Motion prediction using social forces

In the proposed work, the social force model is combined in the prediction step to provide a more accurate prediction of a target motion. It is included as an input function, which is dependent on the other pedestrians and obstacles in the surroundings. Let x^i_{k-1} and x^i_k be the state of the pedestrian ϕ_i at time $k-1$ and k, respectively. The target motion is given by

$$x^i_k = f^i_{k-1}(x^i_{k-1}) + \varpi^i_{k-1}(\phi^i, \mho, O) + \nu^i_{k-1} \tag{8}$$

where f is the state transition density, ϖ is the input component from the social forces and ν is the zero-mean Gaussian noise.

Without the external stimuli, i.e., when $\varpi^i_k(\phi^i, \mho, O)$ is zero, no social forces are exerted and the prediction model reduces to the standard independent motion model prediction. The social force equations are non-linear and no formulas are available for the uncertainty associated with the social force components. Thus, it is not the best to model the uncertainty associated with the social force uncertainty as an additive Gaussian noise component. Hence, the social forces are integrated as an input component. The Particle Filter [3] approach, which describes the posterior density function by a set of random samples with corresponding weights, is suitable for applications with non-linear and non-Gaussian models [3]. However, in a multi-target tracking scenario, the particle filter's computational load is high due to complex data association and the curse of dimensionality [31]. Hence, an SMC implementation and a Gaussian mixture implementation of the proposed SF-PHD filter are given in this paper.

III. Social Force based PHD filter

A. Probability hypothesis density filter

In multitarget tracking with a varying number of targets, it is impractical to use the standard

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

Bayesian formulation in a fixed dimensional space to extract the target states when the number of targets is unknown. Nevertheless, Finite Set Statistics (FISST), a set of practical mathematical tools from point process theory, can be used when the number of targets is time-varying. The PHD filter, which propagates the first moment of the multi-object posterior, which is known as the intensity function or PHD, is a suboptimal, but practical, alternative to the multitarget Bayesian recursive formula within the FISST framework [18].

The integral of probability hypothesis density $D_{k|k}(x_k|Z^k)$ over any area S of the target state space gives the expected number of targets $N_{k|k}$ contained within that area S. That is,

$$N_{k|k} = \int_S D_{k|k}(x_k|Z^k)dx_k \tag{9}$$

where x_k denotes the single-target state space and Z^k denotes all the measurements up to time step k. This unique property that characterizes the PHD is also possessed by the first-order statistical moment of the multitarget posterior distribution. Hence, the first order statistical moment of the multitarget posterior distribution is the PHD [18]. It can be noted that the above definition requires that all higher order moments be neglected so that the multitarget posterior distribution can be approximated at every step by a Poisson point process that is matched to the first-order statistical moment. The cardinality estimate equation (9) is known to be rather unreliable when the number of targets increases and a solution is provided in [19].

The SMC method is a technique to represent the posterior distribution of the probability hypothesis density by a set of random particles. The SMC also includes the state space information with corresponding weights to approximate the PHD [36]. The SMC method maintains a constant ratio between the number of random particles and the expected number of targets by an adaptive allocation of the number of particles. The adaptive mechanism has a significant effect on the computational complexity of the algorithm in that the complexity does not increase exponentially but linearly with the increasing number of targets [26]. The total weight of all the particles gives the expected number of targets at a given time step, which follows from the property that the integral of the PHD over an area gives the expected number of targets.

One of the assumption in the PHD filter is that the target motions are independent across targets. But in the proposed Social Force PHD filter, this assumption is relaxed and the dependency of target motion is included as an input component in the prediction step as discussed in II-A.

B. Sequential Monte Carlo implementation

The proposed RFS-based SMC implementation of Social Force PHD filter is a modified version of the original SMC-PHD filter [36]. As explained earlier, the social forces from the other targets have to be calculated during the prediction process for each target and for this purpose target identities have to be maintained. To keep target identities, a particle label approach is used where each particle in the whole particle set has its own label [38]. This label for the particle is inherited along with the particle evolution procedure and used for state prediction and state extraction.

Let $\{x_{k-1}^{(s)}, w_{k-1}^{(s)}, l_{k-1}^{(s)}\}$ denote the whole particle set at time $k-1$, $x_{k-1}^{(s)}$ and $w_{k-1}^{(s)}$ represent the s^{th} particle and the corresponding weight, $l_{k-1}^{(s)}$ mean the target identity with $l_{k-1}^{(s)} \in \Gamma_{k-1} = \{1, 2, ...\hat{N}_{max}\}$, and $\hat{N}_{max}$ be the maximal target identity until time $k-1$.

Prediction

For $s = 1, ..., L_{k-1}$, generate state samples $\{x_{k|k-1}^{(s)}\}_{s=1}^{L_{k-1}}$ from the proposal density $q(\cdot|x_{k-1}, Z_k)$ and state samples $\{x_{k|k-1}^{(s)}\}_{s=L_{k-1}+1}^{L_{k-1}+J_k}$ corresponding to new spontaneously born targets from another proposal density $p_k(\cdot|Z_k)$. That is,

$$x_{k|k-1}^{(s)} = \begin{cases} q(\cdot|x_{k-1}, Z_k), & s = 1, ..., L_{k-1} \\ p_k(\cdot|Z_k), & s = L_{k-1}+1, ..., L_{k-1}+J_k \end{cases} \tag{10}$$

Then, the predicted PHD can be approximated by the weights as

$$D_{k|k-1}(x_k|Z^{k-1}) = \sum_{s=1}^{L_{k-1}+J_k} w_{k|k-1}^{(s)} \delta(x_k - x_{k|k-1}^{(s)}) \tag{11}$$

where

$$w_{k|k-1}^{(s)} = \begin{cases} \frac{e_{k|k-1}(x_{k-1}^{(s)}) f_{k|k-1}(x_k^{(s)}|X_{k-1}^{(s)}, \mho, O) + b_{k|k-1}(x_{k|k-1}^{(s)}|x_{k-1}^{(s)})}{q_k(x_{k|k-1}^{(s)}|x_{k-1}^{(s)}, Z_k)} w_{k-1}^{(s)}, \\ s = 1, ..., L_{k-1} \\ \frac{\gamma_k(x_{k|k-1}^{(s)})}{p_k(x_{k|k-1}^{(s)}|Z_k)} \frac{1}{J_k}, s = L_{k-1}+1, ..., L_{k-1}+J_k \end{cases}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

where $X_{k-1}^{(s)}$ is the state of all the particles of all the targets at time step $k-1$ and $e_{k|k-1}(x_{k-1}^{(s)})$ is the target existence probability at time step k given it has previous state $x_{k-1}^{(s)}$. The functions that represent the Markov target transition density $f_{k|k-1}(\cdot)$, entry of new targets $\gamma_k(\cdot)$ and target spawning $b_{k|k-1}(\cdot)$ are conditioned on the target motion model. The important difference between the original SMC-PHD filter and the social forces SMC-PHD filter is in the Markov transition density $f_{k|k-1}(\cdot)$, which is given by (8).

Consider a representative scenario where at time step $k-1$ three targets with labels $l_{k-1}^1, l_{k-1}^2, l_{k-1}^3$ exist with ρ particles per target (i.e., there are ρ particles with each label). To predict the state of the target with label l_{k-1}^1, take each particle with label l_{k-1}^1 and find the social force for that particle with all the particles labeled l_{k-1}^2 and particles labeled l_{k-1}^3. Further, the weight of the force from other pedestrians and obstacles $p_k^{i,\kappa}$ in (8) is defined to be proportional to the target existence probability and inversely proportional to the Euclidean distance between the particles. Let i denote a particle labeled l_{k-1}^1 and κ denote a particle labeled l_{k-1}^2 or l_{k-1}^3. Then,

$$p_k^{i,\kappa} = \frac{e_{k|k-1}(x_{k-1}^{(\kappa)})/d_{k-1}^{i,\kappa}}{\sum_\kappa (e_{k|k-1}(x_{k-1}^{(\kappa)})/d_{k-1}^{i,\kappa})} \tag{12}$$

where the existence probability for an obstacle is assumed to be unity.

Since this increases the complexity exponentially with the number of targets, a particle selection technique is carried out before the prediction step. Let the extracted state estimate of target 1 at time step be x_{k-1}. Instead of assigning all the particles labeled l_{k-1}^1 to target 1, only the particles that fall within a validation gate corresponding to target 1 are assigned to target 1. The validation gate can be chosen based on the scenario, e.g., centered at the position components of state x_{k-1} with radius equal to measurement standard deviation. Though the particle selection approach reduces the computational load, the complexity will still increase exponentially with the number of targets, which is a drawback of this algorithm.

Update

The particle weights are updated when the set of measurements Z_k at time step k become available and given by

$$w_k^{*(s)} = \left[(1 - p_D(x_{k|k-1}^{(s)})) + \sum_{i=1}^{N_k^Z} \frac{p_D(x_{k|k-1}^{(s)}) f_{k|k}(z_k^i | x_{k|k-1}^{(s)})}{\lambda_k c_k(z_k^i) + \Psi_k(z_k^i)}\right] w_{k|k-1}^{(s)} \tag{13}$$

where

$$\Psi_k(z_k^i) = \sum_{s=1}^{L_{k-1}+J_k} p_D(x_{k|k-1}^{(s)}) f_{k|k}(z_k^i | x_{k|k-1}^{(s)}) w_{k|k-1}^{(s)} \tag{14}$$

and $f_{k|k}(\cdot)$ is the measurement likelihood function of single target single sensor.

Resample and cluster with label

When we resample targets that are closely-spaced, it is possible that no particle becomes available for a target. This is possible when the measurement uncertainty is relatively high compared to the distances among the nearby targets. This problem arises when consecutive measurements fall away from the actual measurement originating the target resulting in a lower likelihood value. To overcome this problem, a resampling and clustering approach as in [38] is used.

In order for the SMC-SF-PHD filter to work, the social force parameter values need to be known in advance, which is not always realistic. Further, in a multi-target scenario, the social forces between different targets can vary too. In [27], SMC implementation of the Multiple Model PHD (MM-PHD) filter was proposed to track maneuvering targets effectively. A similar approach is used here with the proposed SMC-SF-PHD filter so that each model can be initialized with different social force parameter values. Since the implementation details are the same as in [27], details have been omitted here.

C. Gaussian mixture implementation

The proposed Gaussian mixture implementation of the Social Force PHD filter is a modified version of the original Gaussian mixture filter [37]. Assuming a linear Gaussian multi-target model as in [37], the posterior intensity can be written as a mixture of Gaussian components as follows:

$$D_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} \mathcal{N}(x; m_{k-1}^{(i)}, P_{k-1}^{(i)}) \tag{15}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

where $m_{k-1}^{(i)}$ and $P_{k-1}^{(i)}$ denotes the mean and covariance of ith component of Gaussian mixture, respectively. The predicted intensity is also a Gaussian mixture and at time k it can be written as

$$D_{k|k-1}(x) = D_{S,k|k-1}(x) + \gamma_k(x) \tag{16}$$

where $\gamma_k(x)$ is the birth intensity and $D_{S,k|k-1}(x)$ is given by

$$D_{S,k|k-1}(x) = e_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} \mathcal{N}(x; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}) \tag{17}$$

where $e_{S,k}$ is the target existence probability. Since the target motions are dependent on one another, the mean $m_{S,k|k-1}^{(j)}$ and covariance $P_{S,k|k-1}^{(j)}$ calculations differ from those in the standard Gaussian mixture implementation in [37]. In the proposed Gaussian mixture implementation, the predicted mean and the covariance of each Gaussian component are calculated based on the target motion given by (8). Hence, targets states are stacked and the predicted mean and covariance are derived based on propagating the stacked states and covariances.

To maintain the target's identity, Gaussian components are labeled and propagated to the subsequent time steps as in [23]. Let the total number of targets be n and the stacked state vector at time step $k-1$ be $X_{k-1} = [x_{k-1}^{1,1}, ..., x_{k-1}^{1,m_1}, ..., x_{k-1}^{n,1}, ..., x_{k-1}^{n,m_n}]'$, where x_{k-1}^{i,m_i} represents the state vector of the m_ith Gaussian component of target i. For the targets whose state evolution equations are linear and the motions are dependent on one another, the state equation can be written as

$$X_k = F_{k-1} X_{k-1} + \varpi_{k-1}(\phi, \mho, O) + \nu_{k-1} \tag{18}$$

where

$$F_{k-1} = \text{diag}(F_{k-1}^{1,1}, ..., F_{k-1}^{1,m_1}, ..., F_{k-1}^{n,1}, ..., F_{k-1}^{n,m_n}) \tag{19}$$

$$\varpi_{k-1} = [\varpi_{k-1}^{1,1}, ..., \varpi_{k-1}^{1,m_1}, ..., \varpi_{k-1}^{n,1}, ..., \varpi_{k-1}^{n,m_n}]' \tag{20}$$

$$\nu_{k-1} = [\nu_{k-1}^{1,1}, ..., \nu_{k-1}^{1,m_1}, ..., \nu_{k-1}^{n,1}, ..., \nu_{k-1}^{n,m_n}]' \tag{21}$$

Here, F_{k-1}^{i,m_i} represents the state transition matrix and ν_{k-1}^{i,m_i} represents the process noise of the m_ith Gaussian component of target i. In the above, ν_{k-1}^{i,m_i} is the Gaussian noise with zero mean and covariance Γ_{k-1}^{i,m_i}. Then the covariance matrix of ν_{k-1}, Γ_{k-1}, is given by

$$\Gamma_{k-1} = \text{diag}(\Gamma_{k-1}^{1,1}, ..., \Gamma_{k-1}^{1,m_1}, ..., \Gamma_{k-1}^{n,1}, ..., \Gamma_{k-1}^{n,m_n}) \tag{22}$$

In the case of linear Gaussian dynamics, it can be shown that

$$P_{k|k-1} = [\Gamma_{k-1} + \beth_{k-1} P_{k-1}^{-1} \beth_{k-1}']^{-1} \tag{23}$$

where $\beth_{k-1}$ is derived as in (24) and the stacked covariance matrix P_{k-1} is given by

$$P_{k-1} = \text{diag}(P_{k-1}^{1,1}, ..., P_{k-1}^{1,m_1}, ..., P_{k-1}^{n,1}, ..., P_{k-1}^{n,m_n}) \tag{25}$$

It is assumed in (24) that (1) is differentiable. But this assumption may not hold at cut-offs and numerical differentiation is used in that cases, which is a drawback of the proposed GM-SF-PHD algorithm. Note that a target can have multiple Gaussian components and hence social force components within the same target Gaussian component are zero (see (24)). Further, the predicted covariance $P_{S,k|k-1}^{(j)}$ is the corresponding diagonal block matrix of the stacked predicted covariance matrix $P_{k|k-1}$. The updated state can be calculated using the update equations in [37] but with the stacked observation matrix and stacked measurements. Finally, the state estimates of the individual targets are calculated using the Gaussian components with the same label. When the targets are closely spaced or crossing, the calculation of the final state estimates based on the Gaussian components with the same label may fail. In [23], this problem is addressed using a track management scheme, which uses the past trajectories of the targets to maintain the correct target identity.

To reduce the number of Gaussian components and the computation load, a heuristic pruning algorithm as in [37] is used. In the GM-SF-PHD filter, cross-covariance component calculation is important to find the dependency between the Gaussian components. Hence, using a pruning approach results in poor performance compared to using a pruning approach in an independent motion target scenario. Also, the pruning approach results in loss of information and degrades performance compared to the SMC-SF-PHD filter. Further, the Gaussian components that have weights greater than a threshold value are extracted as target states from the Gaussian representation of the posterior intensity [37].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

$$\urcorner_{k-1} = \begin{bmatrix} F_{k-1}^{1,1} + \frac{\partial \varpi_{k-1}^{1,1}}{\partial x_{k-1}^{1,1}} & \cdots & 0 & \cdots & \cdots & \cdots & \frac{\partial \varpi_{k-1}^{1,1}}{\partial x_{k-1}^{n,m_n}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & F_{k-1}^{1,m_1} + \frac{\partial \varpi_{k-1}^{1,m_1}}{\partial x_{k-1}^{1,m_1}} & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & F_{k-1}^{n,1} + \frac{\partial \varpi_{k-1}^{n,1}}{\partial x_{k-1}^{n,1}} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \varpi_{k-1}^{n,m_n}}{\partial x_{k-1}^{1,1}} & \cdots & \cdots & \cdots & 0 & \cdots & F_{k-1}^{n,m_n} + \frac{\partial \varpi_{k-1}^{n,m_n}}{\partial x_{k-1}^{n,m_n}} \end{bmatrix} \tag{24}$$

During the calculation of $\varpi_{k-1}^{j}(\phi^j, \mho, O)$, the weight of the force from pedestrians and obstacles $p_k^{i,\kappa}$ in (8) needs to be calculated as well. A pedestrian's motion is affected more by another nearby pedestrian than by a far-away pedestrian. The same logic applies to obstacles as well. Hence, the weight $p_k^{i,\kappa}$ is assumed to be inversely proportional to the distance between the targets. Also, target existence probability has to be considered in calculating the weight $p_k^{i,\kappa}$ since a target with low existence probability might be actually a false alarm and its affect on a target should be low compared to a target with high existence probability. Hence, the weight $p_k^{i,\kappa}$ is assumed to be proportional to the existence probability of the other target. Based on the above assumptions, $p_k^{i,\kappa}$ can be written as

$$p_k^{i,\kappa} = \frac{e_{k|k-1}(x_{k-1}^{(\kappa)})/d_{k-1}^{i,\kappa}}{\sum_{\kappa}(e_{k|k-1}(x_{k-1}^{(\kappa)})/d_{k-1}^{i,\kappa})} \tag{26}$$

where the existence probability for an obstacle is assumed to be unity. Note that any other model for weight variation can be used equally well depending on application. Further, multiple social force parameter values can be used to initialize different Gaussian models. The multiple model social force parameter approach is essential in real tracking scenarios where the dependency between the targets is unknown and may vary with time.

IV. PCRLB BOUND FOR DEPENDENT TARGETS

The Posterior Cramer-Rao Lower Bound (PCRLB) [34] is the widely used performance measure in target tracking applications. It gives a lower bound on the error covariance and is defined by $J(k)^{-1}$, where $J(k)$ is the Fisher information matrix (FIM) [5]. Let x_k be an unknown and random state vector, and $\hat{x}_k(Z_k)$ be an unbiased estimate of x_k based on the measurement data Z_k. Then the lower bound of the error covariance matrix is given by

$$\mathbf{E}\left\{[\hat{x}_k - x_k]\ [\hat{x}_k - x_k]'\right\} \geq J(k)^{-1} \tag{27}$$

where $\mathbf{E}$ is the expectation over $\left(x_k, Z_1^k\right)$ and $[\cdot]'$ is the transpose. The posterior FIM $J(k+1)$ can be evaluated recursively as follows:

$$J(k+1) = J_x(k+1) + J_z(k+1) \tag{28}$$

where

$$J_x(k+1) = \mathbf{E}\{-\Delta_{x_{k+1}}^{x_{k+1}} \ln\ p(x_{k+1})\} \tag{29}$$

$$J_z(k+1) = \mathbf{E}\{-\Delta_{x_{k+1}}^{x_{k+1}} \ln\ p(z_{k+1}|x_{k+1})\} \tag{30}$$

and $\Delta_{\varrho}^{\varsigma}$ denotes the partial derivative operator whose (i,j)th component is

$$\Delta_{\varrho}^{\varsigma}(i,j) = \frac{\partial^2}{\partial \varrho(i) \partial \varsigma(j)} \tag{31}$$

and $\varrho(j)$ and $\varsigma(j)$ are the jth components of vectors ϱ and ς, respectively. In the derivation of the standard PCRLB, it is assumed that the motions of targets are independent of each other. In this section, a bound based on the PCRLB for dependent targets, but independent of the proposed SMC-SF-PHD filter and the GM-SF-PHD filter, is derived.

Let the total number of targets be fixed and given by n and the stacked state vector at time step k of all the targets in the surveillance region be $X_k = [x_k^1, x_k^2, ..., x_k^n]'$, where x_k^i represents the state vector of target i. For the targets whose state evolving equations are linear and moving independent of the motion of other targets, the state equation can be written as

$$X_{k+1} = F_k X_k + \nu_k \tag{32}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

where

$$F_k = \text{diag}(F_k^1, F_k^2, ..., F_k^n) \quad (33)$$
$$\nu_k = [\nu_k^1, \nu_k^2, ..., \nu_k^n]' \quad (34)$$

Here, F_k^i represents the state transition matrix and ν_k^i represents the process noise of target i. In the above, ν_k^i is the Gaussian noise with zero mean and covariance Γ_k^i, then the covariance matrix of ν_k, Γ_k, is given by

$$\Gamma_k = [\Gamma_k^1, \Gamma_k^2, ..., \Gamma_k^n]' \quad (35)$$

In the case of linear, Gaussian dynamics, it can be shown that

$$J_x(k+1) = [\Gamma_k + F_k J_x(k)^{-1} F_k']^{-1} \quad (36)$$

In the case of dependent targets where the motion of the targets is given by (8), the new stacked matrix $\beth_k$ can be derived as

$$\beth_k = \begin{bmatrix} F_k^1 + \frac{\partial \varpi_k^1}{\partial x_k^1} & \frac{\partial \varpi_k^1}{\partial x_k^2} & \cdots & \cdots & \frac{\partial \varpi_k^1}{\partial x_k^n} \\ \frac{\partial \varpi_k^2}{\partial x_k^1} & F_k^2 + \frac{\partial \varpi_k^2}{\partial x_k^2} & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \varpi_k^n}{\partial x_k^1} & \cdots & \cdots & \cdots & F_k^n + \frac{\partial \varpi_k^n}{\partial x_k^n} \end{bmatrix} \quad (37)$$

where it is assumed that (1) is differentiable. Finite-difference methods are used when the above assumption fails. Now the $J_x(k+1)$ component for the PCRLB of dependent targets is calculated as follows:

$$J_x(k+1) = [\Gamma_k + \beth_k J_x(k)^{-1} \beth_k']^{-1} \quad (38)$$

The calculation of measurement contribution $J_z(k+1)$ to the PCRLB needs to consider the measurement origin uncertainty, with measurements originating from one of the targets or from clutter. In [33], it is shown that the measurement origin uncertainty can be expressed as an information reduction matrix (IRM) and it is used in this paper as well. It can be noted that the IRM depends on the measurement noise covariance, false alarm rate, field of view of the sensor, probability of detection and distances between the targets in the measurement space.

Consider a representative scenario of two targets where only the social force components are from personal motivation (α_k^i) and repulsion force from the other pedestrians (β_k^i). Based on (8), the motion of the targets can be written as

$$x_{k+1}^1 = F_k^1 x_k^1 + \alpha_k^1(x_k^1) + \beta_k^1(x_k^1, x_k^2) + \nu_k^1 \quad (39)$$
$$x_{k+1}^2 = F_k^2 x_k^2 + \alpha_k^2(x_k^2) + \beta_k^2(x_k^1, x_k^2) + \nu_k^2 \quad (40)$$

Now the stacked transition matrix can be written as

$$\beth_k = \begin{bmatrix} F_k^1 + \frac{\partial \alpha_k^1(x_k^1)}{\partial x_k^1} + \frac{\partial \beta_k^1(x_k^1, x_k^2)}{\partial x_k^1} & \frac{\partial \beta_k^1(x_k^1, x_k^2)}{\partial x_k^2} \\ \frac{\partial \beta_k^2(x_k^1, x_k^2)}{\partial x_k^1} & F_k^2 + \frac{\partial \alpha_k^2(x_k^1)}{\partial x_k^2} + \frac{\partial \beta_k^2(x_k^1, x_k^2)}{\partial x_k^2} \end{bmatrix} \quad (41)$$

The states of the targets are $x^1 = [\xi_1, \dot{\xi}_1, \eta_1, \dot{\eta}_1]'$ and $x^2 = [\xi_2, \dot{\xi}_2, \eta_2, \dot{\eta}_2]'$, where (ξ_1, η_1) are the position components of target 1, $(\dot{\xi}_1, \dot{\eta}_1)$ denote the velocity components of target 1 at time step k. The targets are moving towards a destination at (ξ_d, η_d). In order to simplify the representation, the time step information k is omitted in the above notation. It is assumed that the magnitude parameter of the exerted force from the other target in (4) $a^1 = a^2 = a$, the range parameter of the force in (4) $b^1 = b^2 = b$ and the sampling time interval is T. Since there are only two targets, weight of the social forces $p_k^{12} = p_k^{21} = 1$. In order to simplify the representation, the following notation is used:

$$\begin{aligned} \Omega &= \frac{\partial \alpha_k^1(x_k^1)}{\partial x_k^1} \\ \Upsilon &= \frac{\partial \beta_k^1(x_k^1, x_k^2)}{\partial x_k^1} \\ \chi &= \left((\xi_d - \xi_1)^2 + (\eta_d - \eta_1)^2\right)^{3/2} \\ \omega &= \sqrt{((\xi_1 - \xi_2)^2 + (\eta_1 - \eta_2)^2)} \\ \zeta &= \left((\xi_1 - \xi_2)^2 + (\eta_1 - \eta_2)^2\right)^{3/2} \mathbf{e}^{\frac{\sqrt{(\xi_1 - \xi_2)^2 + (\eta_1 - \eta_2)^2} - (r_1 + r_2)}{b_2}} \end{aligned} \quad (42)$$

The elements of the 4x4 matrix Ω can be found as follows:

$$\Omega_{11} = -\frac{m_1 \bar{v} T^2}{2\tau} \frac{(\eta_d - \eta_1)^2}{\chi} \quad (43)$$
$$\Omega_{12} = -\frac{m_1 T^2}{2\tau} = \Omega_{34} \quad (44)$$
$$\Omega_{13} = -\frac{m_1 \bar{v} T^2}{2\tau} \frac{(\xi_d - \xi_1)(\eta_d - \eta_1)}{\chi} = -\Omega_{31} \quad (45)$$
$$\Omega_{21} = -\frac{m_1 \bar{v} T}{\tau} \frac{(\eta_d - \eta_2)^2}{\chi} \quad (46)$$
$$\Omega_{22} = -\frac{m_1 T}{\tau} = \Omega_{44} \quad (47)$$
$$\Omega_{23} = -\frac{m_1 \bar{v} T}{\tau} \frac{(\xi_d - \xi_2)(\eta_d - \eta_2)}{\chi} = -\Omega_{41} \quad (48)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

$$\Omega_{33} = -\frac{m_1 \bar{v} T^2}{2\tau} \frac{(\xi_d - \xi_2)^2}{\chi} \quad (49)$$

$$\Omega_{43} = -\frac{m_1 \bar{v} T}{\tau} \frac{(\xi_d - \xi_2)^2}{\chi} \quad (50)$$

$$\Omega_{ij} = 0 \qquad \text{otherwise} \quad (51)$$

The elements of the 4x4 matrix Υ as follows:

$$\Upsilon_{11} = \frac{aT^2\left((\xi_1 - \xi_2)^2\omega - b_2(\eta_1 - \eta_2)^2\right)}{2\ b\ \zeta} \quad (52)$$

$$\Upsilon_{13} = \frac{aT^2(\xi_1 - \xi_2)(\eta_1 - \eta_2)(b + \omega)}{2\ b\ \zeta} = \Upsilon_{31} \quad (53)$$

$$\Upsilon_{21} = \frac{aT\left((\xi_1 - \xi_2)^2\omega - b_2(\eta_1 - \eta_2)^2\right)}{b\ \zeta} \quad (54)$$

$$\Upsilon_{23} = \frac{aT(\xi_1 - \xi_2)(\eta_1 - \eta_2)(b + \omega)}{b\ \zeta} = \Upsilon_{41} \quad (55)$$

$$\Upsilon_{33} = \frac{aT^2\left((\eta_1 - \eta_2)^2\omega - b_2(\eta_1 - \eta_2)^2\right)}{2\ b\ \zeta} \quad (56)$$

$$\Upsilon_{43} = \frac{aT\left((\eta_1 - \eta_2)^2\omega - b_2(\eta_1 - \eta_2)^2\right)}{b\ \zeta} \quad (57)$$

$$\Upsilon_{ij} = 0 \qquad \text{otherwise} \quad (58)$$

It can be shown that for a two-target scenario, $\frac{\partial \beta_k^1(x_k^1, x_k^2)}{\partial x_k^2} = -\Upsilon$. Similarly, the elements of matrices $\frac{\partial \alpha_k^2(x_k^2)}{\partial x_k^2}$, $\frac{\partial \beta_k^2(x_k^1, x_k^2)}{\partial x_k^2}$, $\frac{\partial \beta_k^2(x_k^1, x_k^2)}{\partial x_k^1}$ can be calculated as well.

V. Results with simulated and real data

A. Simulation

In this section, two-dimensional tracking simulations are presented to compare the SMC-SF-PHD filter and GM-SF-PHD filter's performance with those of other tracking algorithms. The targets move in the surveillance region [350,550] m $\times$ [350,550] m. The target state takes the form $x_k = [\xi_k, \dot{\xi}_k, \eta_k, \dot{\eta}_k]'$, where ξ_k and η_k are the position components, $\dot{\xi}_k$ and $\dot{\eta}_k$ denote the velocity components of the target state at time step k. The motion of target ϕ^i is given by

$$x_k^i = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1}^i + \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \varpi_{k-1}^i(\phi^i, \mho) + \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} v_{k-1}^i \quad (59)$$

where the sampling interval is $T = 1$s. For the simulations, the social force component from personal motivation (2) and the repulsive force component from other targets (4) are considered. However, the other social forces can be integrated into the proposed framework without further theoretical modification. The following social force parameters are used:

- Radius of a target (r^i) = 0.2 m
- Mass of a target (m^i) = 80 kg
- Desired velocity ($\bar{v}^i$) = 2 m/s
- Relaxation time (τ^i) = 1 s
- Magnitude parameter of the force from the other target (a^κ) = 70 N
- Range parameter of the force (b^κ) = 1 m

Initially, the same parameters are used in the proposed tracking method. Subsequently, slightly different values are used to ensure that the proposed method is insensitive to minor parameter uncertainties. A statistically robust algorithm that can estimate those parameters online is in the works.

The sensor measurement consists of the targets' position in X and Y direction resulting in observation matrix

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (60)$$

The sensor measurement error covariance is 2m in both X and Y positions. The target detection probability is 0.8 and the false alarm density is 10^{-5} m^{-2} with uniform spatial distribution over the surveillance region. The number of false alarms is Poisson-distributed. The survival probability $e_{k|k-1}$ for each existing target is 0.95 and is independent of the target's state. The number of particles for each existing target and new born target is set to 500.

The Root Mean Squared Error (RMSE) values for position, and velocity estimates, the Optimal Subpattern Assignment (OSPA) [29] and the proposed PCRLB for dependent targets are used to evaluate performance. The measurements are supported by 100 MC runs performed on the same target trajectories but with independently generated measurements for each run.

The proposed SMC-SF-PHD filter and GM-SF-PHD filter are compared against the standard SMC-PHD filter [36], SMC-MTB filter [28], Multiple Model Social-Force Multiple Hypothesis Tracker (MM-SF-MHT) and the Multiple Model Social-Force Joint Integrated Probabilistic Data Association (MM-SF-JIPDA) tracker. The standard SMC-PHD filter implementation differs from the proposed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

filter in the input component where the social force component is not integrated in the SMC-PHD filter. Apart from that, the implementations of the SMC-SF-PHD filter and SMC-PHD filter are the same including particle labeling and label-based resampling. The MM-SF-MHT tracker and MM-SF-JIPDA tracker implementations are different from the standard IMM-MHT [8] tracker and IMM-JIPDA [21] tracker, respectively in the prediction step. In both the MM-SF-MHT and MM-SF-JIPDA trackers, prediction is carried out based on (59) for the calculation of the mean value. The covariance Ξ_k^i of target ϕ^i at time step k is calculated as follows:

$$\Xi_k^i = F_{k-1}^i \Xi_{k-1} (F_{k-1}^i)' + Q_{k-1}^i \tag{61}$$

where F_{k-1}^i is the state transition matrix and Q_{k-1}^i is the covariance matrix of v_{k-1}^i. The covariance calculation in (61) is similar to the approach in [16].

1) Simulation 1: In this simulation, two targets, namely, target 1 and target 2, are moving with initial positions (501, 400) m and (503, 400) m, respectively, and with initial velocities (-0.2, 1.5) m/s and (-0.4, 1.6) m/s, respectively. Both targets are moving towards a destination at (500, 500) m and their motions are dependent on each other. In generating the truth, the process noise component for the target motion are $\sigma_m = 0.005$ m/s^2 in the X direction and $\sigma_m = 0.05$ m/s^2 in the Y direction. The same values are used in the SMC-SF-PHD filter, GM-SF-PHD filter and the SMC-PHD filter. For the IMM estimators [9] in the MM-SF-MHT tracker and the MM-SF-JIPDA tracker, two second-order white noise acceleration models [5] are used. For the first model, noise variances of 0.005 m/s^2 and 0.05 m/s^2 in the X directions and Y direction are used, respectively. For the second model, noise variances of 0.025 m/s^2 and 0.25 m/s^2 in the X direction and Y direction are used, respectively. The position ground truth of the two tracks over 50s are shown in Fig. 1.

2) Simulation 2: In this simulation, seven targets that are traveling towards the destination at (500, 500)m are considered. While generating the ground truth, the process noise component for target motion are $\sigma_m = 0.1$ m/s^2 in the X direction and $\sigma_m = 0.1$ m/s^2 in the Y direction. The same values are used for the SMC-SF-PHD filter, GM-SF-PHD filter and SMC-PHD filter. For the IMM estimators in the MM-SF-MHT tracker and the MM-SF-JIPDA tracker, two second-order white noise acceleration models [5] are used. For the first model, noise variances of 0.1 m/s^2 and 0.1 m/s^2 in the X direction and Y direction are used, respectively. For the second model, noise variances of 0.25 m/s^2 and 0.25 m/s^2 in the X direction and Y direction are used, respectively. The positions of the targets over 45s are shown in Fig. 2. Fig. 3 shows the varying number of actual targets and estimated targets from different algorithms over time. The number of targets is estimated over 100 Monte Carlo runs. It is calculated as the sum of the particle weights and sum of the weights of Gaussian components for the SMC-SF-PHD filter and the GM-SF-PHD filter, respectively.

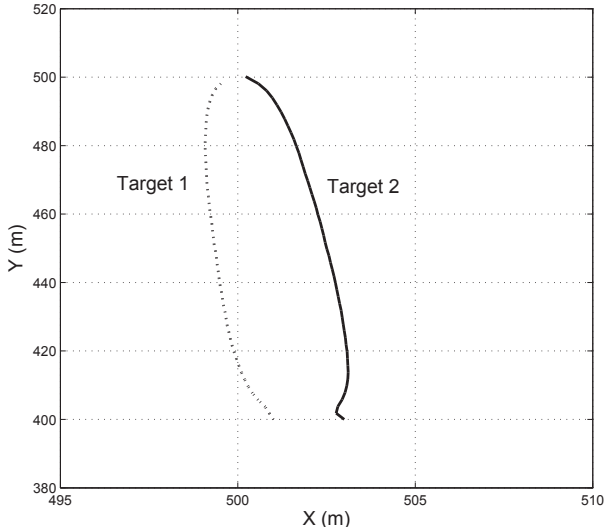

Fig. 1: Ground truth over 50 time steps

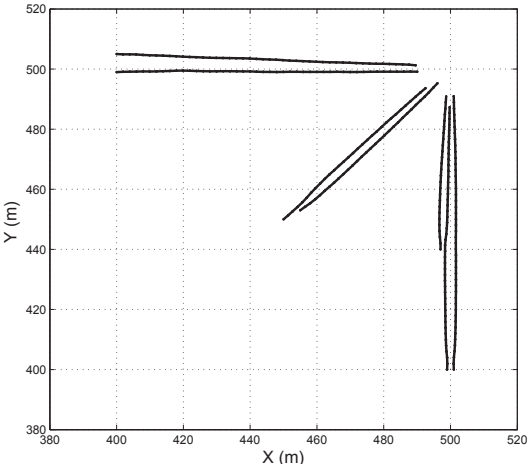

Fig. 2: Seven target trajectories over 45 time steps

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

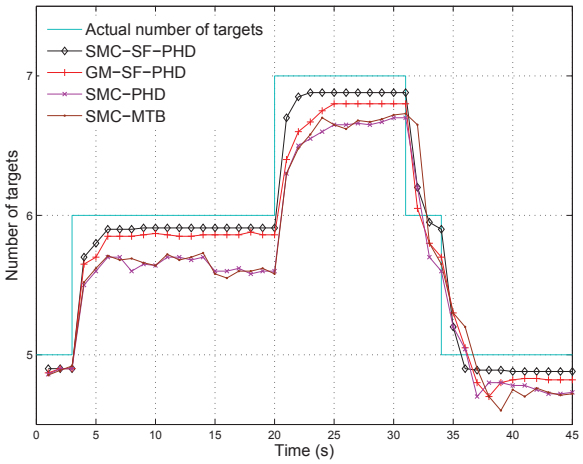

Fig. 3: Number of targets over 45 time steps

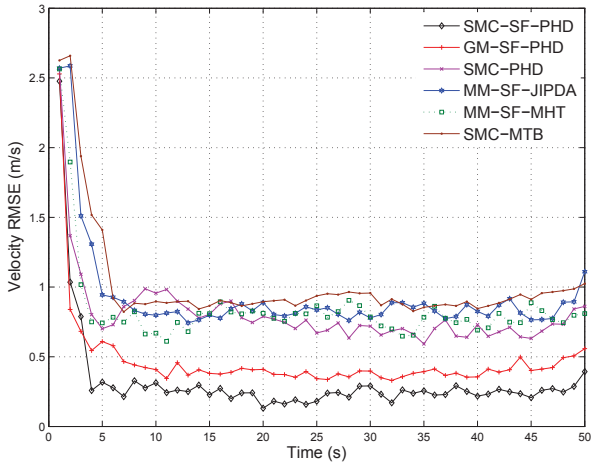

Fig. 5: RMS velocity errors (100 Monte Carlo runs)

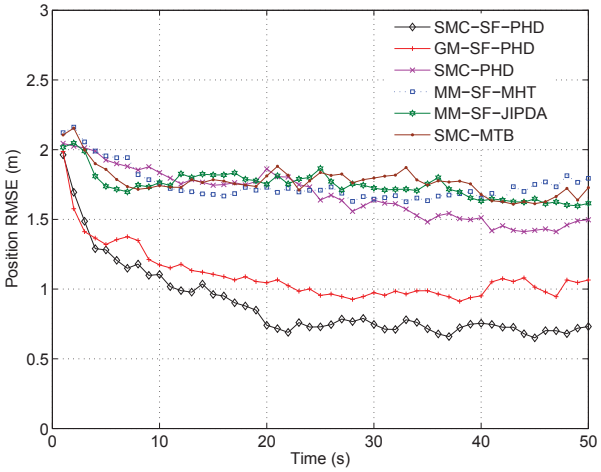

Fig. 4: RMS position errors (100 Monte Carlo runs)

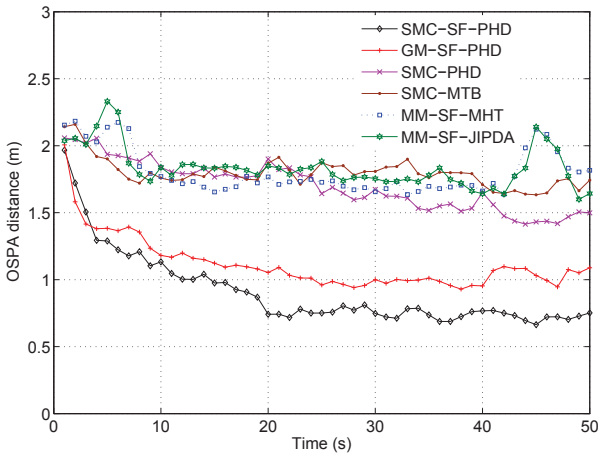

Fig. 6: OSPA distance (100 Monte Carlo runs)

3) Simulation Discussion: Fig. 4 and Fig. 8 show the RMSE in position for simulation 1 and simulation 2, respectively. The proposed SMC-SF-PHD filter outperforms the other filters in both scenarios. Due to the repulsion force between the targets, it can be seen that the targets are keeping a distance between them as they evolve. The SMC-SF-PHD has the advantage over the standard SMC-PHD filter as the repulsion force is modeled through the social force input component. As a result of the SMC-PHD filter's mismatched prediction model, high likelihood values are calculated when a target's predicted state is associated to a measurement from another target. This results in the poor performance

TABLE I: Average performance metrics

Algorithm	RMSE position (m)	RMSE velocity (m/s)	OSPA distance (m)	CPU time (Normalized)
SF-SMC-PHD	0.87	0.32	0.91	21.4
GM-SF-PHD	1.09	0.47	1.12	0.43
SMC-PHD	1.67	0.80	1.71	1
MM-SF-MHT	1.75	0.92	1.80	2.51
MM-SF-JIPDA	1.74	0.86	1.83	1.1
SMC-MTB	1.77	0.99	1.81	1.14

TABLE II: Average performance metrics (multiple model)

Algorithm	RMSE position (m)	RMSE velocity (m/s)
SF-SMC-PHD	1.02	0.41
GM-SF-PHD	1.35	0.59
SMC-PHD	2.15	0.99
MM-SF-MHT	2.35	1.17
MM-SF-JIPDA	2.37	1.19
SMC-MTB	2.21	1.17

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

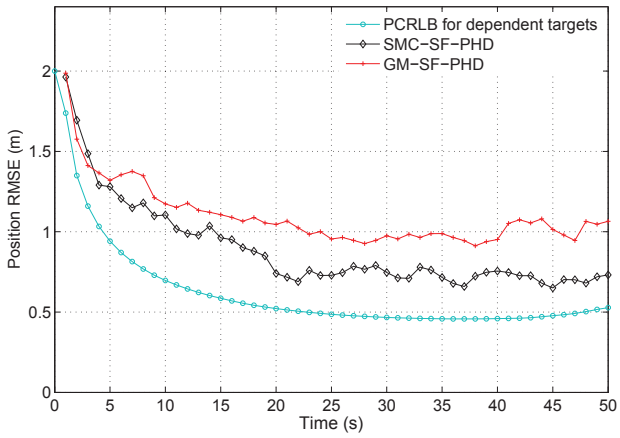

Fig. 7: PCRLB for dependent targets and RMSE

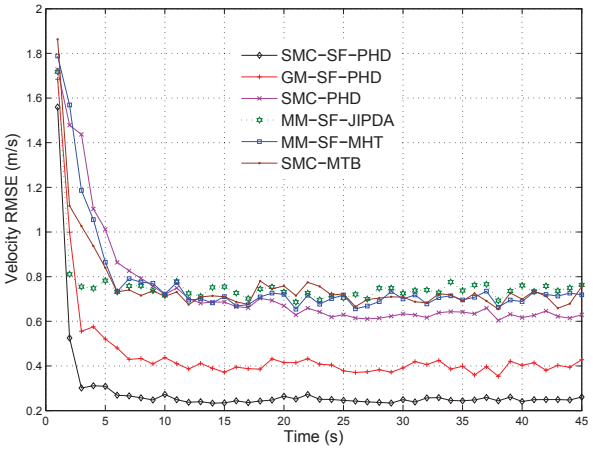

Fig. 9: RMS velocity errors (100 Monte Carlo runs)

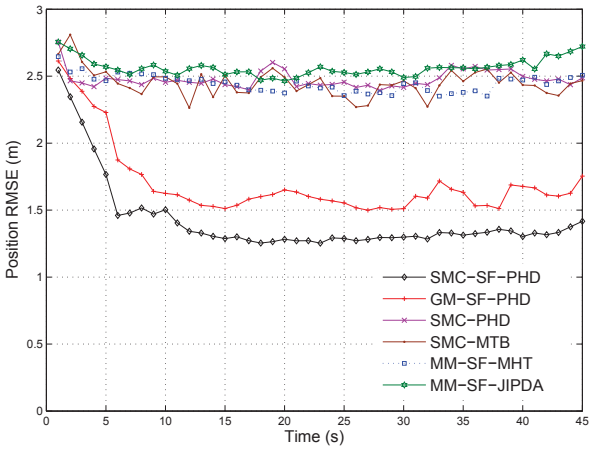

Fig. 8: RMS position errors (100 Monte Carlo runs)

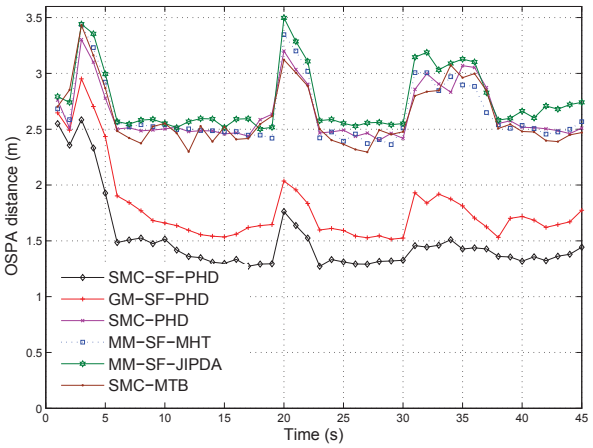

Fig. 10: OSPA distance (100 Monte Carlo runs)

of the standard SMC-PHD filter compared to the proposed SMC-SF-PHD filter. The SMC-MTB filter includes the social force model as an additional weighting factor but it does not factor in the social force in an explicit manner to predict the state. Hence, the performance improvement from the SMC-MTB filter is marginal compared to that in the SMC-PHD filter. The MM-SF-MHT and MM-SF-JIPDA trackers are implemented using the EKF in their IMM implementation. The EKF requires proper error propagation (covariance) mechanism. The covariance calculation in the above two trackers are based on (61) where correlated covariance terms across two targets are not included and a stack-based approach is not considered in the subsequent time steps. In these two trackers, the covariance is initial-

TABLE III: Average performance metrics

Algorithm	RMSE position (m)	RMSE velocity (m/s)	OSPA distance (m)	CPU time (Normalized)
SMC-SF-PHD	1.42	0.28	1.50	51.3
GM-SF-PHD	1.70	0.43	1.79	0.37
SMC-PHD	2.47	0.74	2.70	1
MM-SF-MHT	2.44	0.77	2.75	2.02
MM-SF-JIPDA	2.51	0.76	2.77	0.77
SMC-MTB	2.47	0.75	2.72	1.35

TABLE IV: Average performance metrics (multiple model)

Algorithm	RMSE position (m)	RMSE velocity (m/s)
SMC-SF-PHD	1.65	0.45
GM-SF-PHD	1.97	0.65
SMC-PHD	2.95	0.89
MM-SF-MHT	3.15	1.25
MM-SF-JIPDA	3.26	1.21
SMC-MTB	3.17	1.15

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

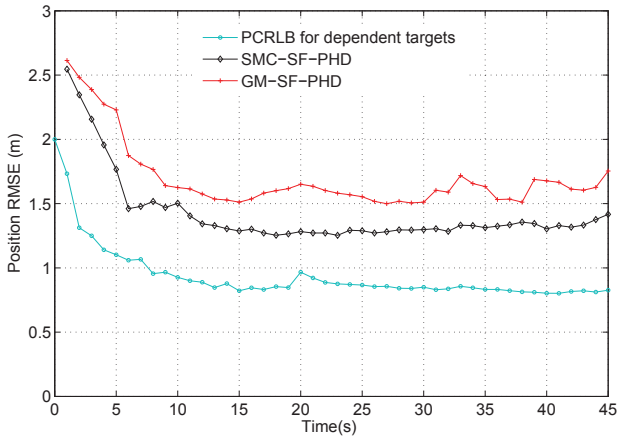

Fig. 11: PCRLB for dependent targets and RMSE

ized based on the surveillance region for position and based on the maximum velocity for velocity [5]. In the subsequent time steps, measurements from another target might fall into the predicted covariance ellipsis of a target being considered. This results in poor data association and in the consecutive time steps leads to degraded social force input component calculation too. As a result, these two trackers' prediction performance degrades and yields poorer accuracies compared to that of the SMC-SF-PHD filter. The covariance calculation in the GM-SF-PHD filter is based on the equations derived in III-C where target motion dependency is included in the state transition matrix and propagated to the subsequent time steps using the stack-based approach. Hence, the proposed GM-SF-PHD filter performs better than the MM-SF-MHT and MM-SF-JIPDA trackers. It is also observed that the SMC-SF-PHD and GM-SF-PHD filters yield almost no false tracks compared to the other filters. Further tests are carried out with a single white noise acceleration model, instead of using two second-order white noise acceleration models in the MM-SF-MHT tracker and the MM-SF-JIPDA tracker. It is observed that the MM-SF-MHT tracker and the MM-SF-JIPDA trackers perform slightly better with two noise models compared to using a single noise model.

Fig. 5 and Fig. 9 show the RMSE in velocity for the two simulations. It is observed that the SMC-SF-PHD filter's performance is better than those of the other filters mainly due to the integrated motivational force component. Though the GM-SF-PHD filter, MM-SF-MHT tracker and the MM-SF-JIPDA trackers also have the social force input component, the advantage of the particle based approach in the proposed SMC-SF-PHD filter contributes to the latter algorithm's better performance.

Fig. 6 and Fig. 10 show the OSPA distance ($p = 2$ and $c = 10$) vs. time for the two simulations. It can be seen that the trend in the OSPA is almost the same as that in the position RMSE metric except when there is a change in the number of targets. It is observed that the proposed algorithms are almost free of false tracks, hence they performed relatively better even when the targets are entering or leaving the surveillance region. The proposed PCRLB for dependent targets is used to validate the performance of the proposed SMC-SF-PHD filter and the GM-SF-PHD filter. For simulation 1, (44)–(58) can be used directly. Fig 7 and Fig. 11, show the proposed PCRLB bound for the dependent targets compared to the proposed filters' RMSE of position estimates. One can see that the filter calculated values follow the PCRLB values.

Table I and Table III show the average performance metrics for simulation 1 and simulation 2, respectively. Based on the simulation results, one can conclude that the SMC-SF-PHD and the GM-SF-PHD filter perform better than the other algorithms. Note that the difference in performance improvement among the other four algorithms is not significant. The last column of Table I and Table III shows the normalized total execution time taken by the CPU for each filter. It can be observed that the GM-SF-PHD is the fastest among the filters compared above. Further, the SMC-SF-PHD is the slowest due to the social force calculation among the particles.

TABLE V: Social force parameter values

Social force parameter	Model 1	Model 2
Desired initial velocity (m/s)	1.5	2.5
Magnitude parameter of the force from the other target (N)	40	110
Range parameter of the force (m)	2	1
Relaxation time (s)	1	1
Mass of a target (kg)	100	70
Radius of a target (m)	0.2	0.2

Table II and IV show the average performance metrics where the multiple model social force parameter values are used for tracking. Table V shows the social force parameter values of the two models used in this simulation. Results show that the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

multiple model versions of the proposed SMC-SF-PHD and GM-SF-PHD filters perform better than the other algorithms. But the performance is still worse than what is possible while using the actual values. It is not practical to know the social force parameters beforehand and using the multiple model approach is a viable solution at the expense of some performance degradation compared with the ideal performance. Also, with a variable-structure multiple model approach, entirely different types of pedestrian motions can be handled in a manner analogous to [15].

B. Real data

To validate the proposed algorithms with real data, experiments are carried out with the data set from [1]. Since the social force parameter values are unknown, two sets of social force parameter values are used with multiple models in all filters. The values of the two sets of social force parameters are assigned based on some calibration trials as follows:

TABLE VI: Social force parameter values

Social force parameter	Model 1	Model 2
Desired initial velocity (m/s)	1	2
Magnitude parameter of the force from the other target (N)	20	100
Range parameter of the force (m)	1	0.5
Relaxation time (s)	0.6	0.6
Radius of a target (m)	0.2	0.2
Mass of a target (kg)	80	80

The data set has 179 frames at 25 frames per second and totally nine pedestrians enter and leave the sensor's field of view at different times. Fig. 12 shows the pedestrian trajectories in the scenario. The measurements are generated every 15 frames by a sensor with H matrix as in (60) and a measurement error standard deviation of 0.5m in both X and Y positions. Since the destination of each target is unknown, the virtual goal approach [16] is used with a goal ahead time of 3 scans. Also, the intended velocity is initialized with the listed values above in Table VI and later updated with the average velocity of the last 3 scans for each pedestrian.

It is observed that proposed algorithms outperformed the other algorithms for Target 3 and Target 4, which seem to have motion correlation, and performed almost same for the other targets. Fig. 13 Fig. 14 show the tracks created by the GM-SF-PHD algorithm and MM-SF-MHT algorithm,

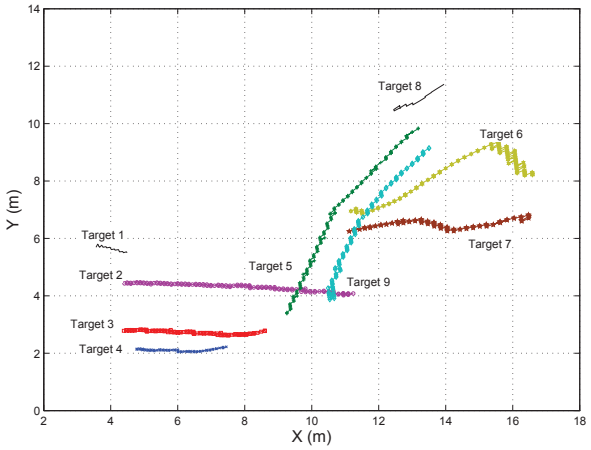

Fig. 12: Pedestrian trajectories

respectively. The better performance of the GM-SF-PHD algorithm in Fig. 13 is due to the social force modeling of repulsion force that exist between Target 3 and Target 4. The MM-SF-MHT and MM-SF-JIPDA algorithms do not have a proper covariance propagation mechanism, hence their tracks are lost and false tracks are created as shown in Fig. 14. The estimates given by the SMC-SF-PHD and GM-SF-PHD algorithms are almost free of false tracks compared to the other algorithms. Fig. 15 and Fig. 16 show the average RMSE and OSPA distance of all the targets in position for the real data test.

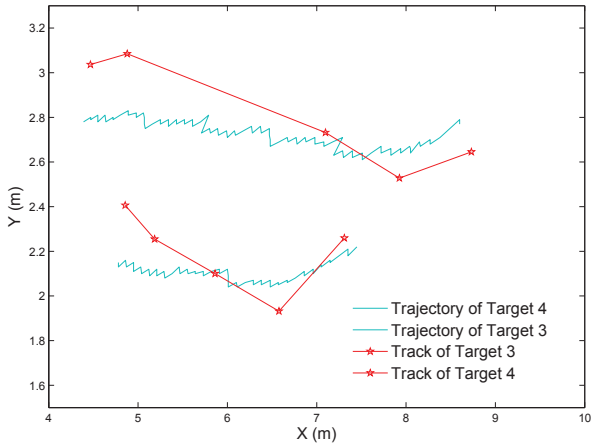

Fig. 13: Tracks from GM-SF-PHD algorithm

VI. Conclusions

In this paper, a new approach was proposed to track the motion of pedestrians whose motion is dependent on one another. It was assumed that a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2017.2680718, IEEE Transactions on Aerospace and Electronic Systems

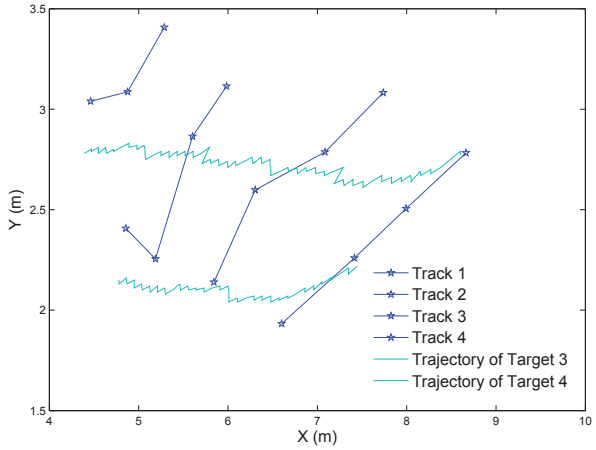

Fig. 14: Tracks from MM-SF-MHT algorithm

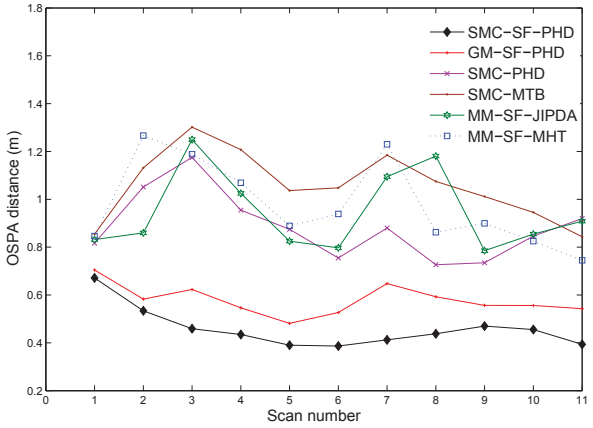

Fig. 16: OSPA distance of all the targets

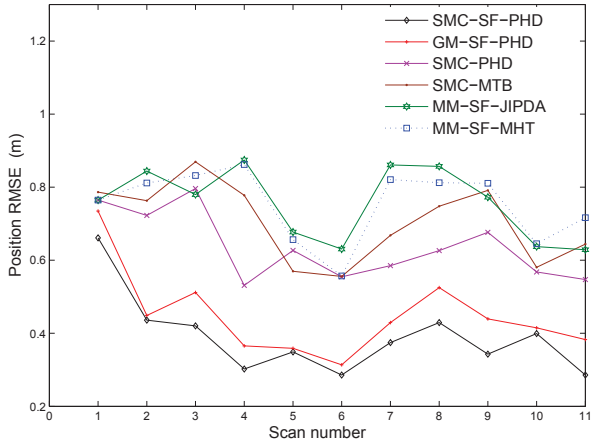

Fig. 15: RMS position errors of all the targets

target's motion is influenced by desired destination, other targets' motion, objects in the surroundings, desired destination and other physical constraints. In existing literature, the constant velocity and constant acceleration models with zero-mean Gaussian process noise are typically used to model the motion of ground targets. In our work, the social force concept integrated with the PHD filter was proposed to model sophisticated target motion possibilities. The social force concept, developed to model pedestrian motion, was mathematically integrated as an input component in the prediction process. To ensure the applicability of the SF-PHD filter, an SMC approach and a Gaussian mixture approach were developed. In the SMC implementation, a target's identity was maintained in the implementation by using the particle labeling approach. Similarly, in the Gaussian mixture approach, a target's identity was maintained using labels for the Gaussian components. Further, to evaluate the performance of the proposed PHD filters, an error bound independent of the proposed SMC-SF-PHD filter and the GM-SF-PHD filter is developed based on the PCRLB taking into account inter-dependent target state evolution. Tracking results using simulated and real data showed that the proposed filters perform better than the SMC-PHD filter, SMC-MTB filter, MM-SF-JIPDA tracker and the MM-SF-MHT tracker. Further, multiple model versions of the proposed filters were used when the actual social force parameter values were unknown in advance. The newly-developed PCRLB for dependent targets was used to validate the performances of the SMC-SF-PHD filter and the GM-SF-PHD filter. It is also observed that SMC-SF-PHD filter outperforms the GM-SF-PHD filter at the expense of computation time. Future work will consider more realistic scenarios where social force parameters such as desired velocity and relaxation time are unknown and need to be estimated online.

REFERENCES

[1] Andriluka, M., Roth, S., and Roth, S.
Monocular 3D pose estimation and tracking by detection.
IEEE Conference on Computer Vision and Pattern Recognition, (June 2010) 623–630

[2] Antonini, G., Martinez, S., Bierlaire, M., and Thiran, J.
Behavioral priors for detection and tracking of pedestrians in video sequences.
International Journal of Computer Vision, **69**, 2 (2006) 159–180

[3] Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking.
IEEE Transactions on Signal Processing, **50**, 2 (February 2002) 174–188

[4] Bar-Shalom, Y., Willet, P.K, and Tian, X.
Tracking and Data Fusion: A Handbook of Algorithms.
YBS Publishing, Storrs, CT, 2011.

[5] Bar-Shalom, Y., Li, X.R., and Kirubarajan, T.
Estimation With Applications On Tracking And Navigation.
Storrs, CT: YBS Publishing, 2001

[6] Bierlaire, M., Antonini, G., and Weber, M.
Behavioral dynamics for pedestrians.
10th International Conference on Travel Behavior Research, (August 2003)

[7] Blackman, S.S., Busch, M. T., and Popoli, R.F.
IMM/MHT tracking and data association for benchmark tracking problem.
IEEE Transactions on American Control Conference, **4**, 1 (June 1995) 2606–2610

[8] Blackman, S.S.
Multiple hypothesis tracking for multiple target tracking.
IEEE Transactions on Aerospace and Electronics Systems, **19**, 1 (January 2004) 5–18

[9] Blom, H.A.P., and Bar-Shalom, Y.
The interacting multiple model algorithm for systems with Markovian switching coefficients.
IEEE Transactions on Automatic Control, **33**, 8 (August, 1988), 780–783

[10] Canaud, M., Mihaylova, L., Faouzi, N.E.E., Billot, R., and Sau, J.
A probabilistic hypothesis density filter for traffic flow estimation in the presence of clutter.
Workshop on Sensor Data Fusion: Trends, Solutions, Applications (SDF), **19**, 1 (2012) 31–36

[11] Colombo, R.M., Goatin, P., Maternini, G., and Rosini, M.D.
Macroscopic models for pedestrian flows.
Proceedings of Big Events and Transport, (October, 2010)

[12] Gipps, P.G, and Marksjo, B.
A micro-simulation model for pedestrian flows.
Mathematics and computers in simulation, **27**, (1985), 95–105

[13] Helbing, D., and Molnár, P.
Social force model for pedestrian dynamics.
American Physical Society, **51**, (May, 1995), 4282–4286

[14] Helbing, D., Molnar, P., Farkas, I.J., and Bolay, K.
Self-organizing pedestrian movement.
Environment and Planning B: Planning and Design, **28**, (2001), 361–383

[15] Kirubarajan, T., Bar-Shalom, Y., Pattipati, K.R., and Kadar, I.
Ground target tracking with variable structure IMM estimator.
IEEE Transactions on Aerospace and Electronic Systems, **36**, 1 (January, 2000), 26–46.

[16] Luber, M., Stork, J.A., Tipaldi, G.D., and Arras, K.O.
People tracking with human motion predictions from social forces.
IEEE International Conference on Robotics and Automation, (May, 2010), 464–469

[17] Luber, M., Tipaldi, G., and Arra. K.
Better models for people tracking.
IEEE International Conference on Robotics and Automation, (2011)

[18] Mahler, R.P.S.
Multitarget Bayes filtering via first-order multitarget moments.
IEEE Transactions on Aerospace and Electronics Systems, **39**, 4 (October, 2003), 1152–1178

[19] Mahler, R.P.S.
PHD filters of higher order in target number.
IEEE Transactions on Aerospace and Electronics Systems, **43**, 4 (October, 2007), 1523–1543

[20] Mahler, R.P.S.
Urban multitarget tracking via gas-kinetic dynamics models.
Proceedings of SPIE, Signal Processing, Sensor Fusion, and Target Recognition XXII, **8745**, (May, 2013), 1–8

[21] Musicki, D., and Suvorova, S.
Tracking in clutter using IMM-IPDA based algorithms.
IEEE Transactions on Aerospace and Electronics Systems, **44**, 4 (January, 2007), 111–127

[22] Okazaki, S.
A study of pedestrian movement in architecture space, Part 1: Pedestrian movement by the application on magnetic models.
Journal of Architecture, Planning. Environment Engineering, **429**, (1979), 51–59

[23] Panta, K., Clark, D.E., and Vo, B.N.
Data association and track management for the Gaussian mixture probability hypothesis density filter.
IEEE Transactions on Aerospace and Electronic Systems, **45**, (July, 2009), 1003–1016

[24] Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L.
You'll never walk alone: Modeling social behavior for multitarget tracking.
IEEE International Conference on Computer Vision, (September, 2009), 261–268

[25] Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L.
Wrong turn - No dead end: A stochastic pedestrian motion model.
IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), (June, 2010), 15–22

[26] Punithakumar, K., Kirubarajan, T., and Sinha, A.
A sequential Monte Carlo probability hypothesis density algorithm for multitarget track before detect.
Proceedings of SPIE , **5913**, (2005), 1–8

[27] Punithakumar, K., Kirubarajan, T., and Sinha, A.
Multiple-model probability hypothesis density filter for tracking maneuvering targets.
IEEE Transactions on Aerospace and Electronic Systems, **44**, 1 (January 2008), 87–98

[28] Reuter, S., and Dietmayer, K.
Pedestrian tracking using random finite sets.
14th International Conference on Information Fusion, (July, 2011), 1–8

[29] Schuhmacher, D., Vo, B.T., and Vo, B.N.
A consistent metric for performance evaluation of multi-object filters.
IEEE Transactions on Signal Processing, **56**, 8 (August, 2008), 3447–3457

[30] Streller, D
Road map assisted ground target tracking.
11th International Conference on Information Fusion, (July, 2008), 1–7

[31] Sutharsan, S., Kirubarajan, T., Lang, T., and Mcdonald, M.
An optimization-based parallel particle filter for multitarget tracking.
IEEE Transactions on Aerospace and Electronic Systems, **48**, 2 (April, 2012), 1601–1618

[32] Teknomo, K., Takeyama, Y., and Inamura, H.
Review on microscopic pedestrian simulation model.
Proceedings of Japan Society of Civil Engineering Conference, (2000).

[33] Tharmarasa, R., Kirubarajan, T., Hernandez, M.L., and Sinha, A.
PCRLB-based multisensor array management for multitarget tracking.
IEEE Transactions on Aerospace and Electronic Systems, **43**, 2 (April, 2007), 539–555.

[34] Tichavsky, P., Muravchik, C.H., and Nehorai, A.

Posterior Cramer-Rao bounds for discrete-time nonlinear filtering.
IEEE Transactions on Signal Processing, **46**, 5 (May, 1998), 1386–1396.

[35] Ulmke, M., and Koch, W.
Road-map assisted ground moving target tracking.
IEEE Transactions on Aerospace and Electronic Systems, **42**, 4 (October, 2006), 1264–1274.

[36] Vo, B.N., Singh, S., and Doucet, A.
Sequential Monte Carlo methods for multi-target filtering with random finite sets.
IEEE Transactions on Aerospace and Electronics Systems , **41**, (2005), 1224–1245

[37] Vo, B.N., and Ma, W.
The Gaussian mixture probability hypothesis density filter.
IEEE Transactions on Signal Processing, (2006), 4091–4104

[38] Zhu, H., Han, C., and Lin, Y.
Particle labeling PHD filter for multi-target track-valued estimates.
14th International Conference on Information Fusion, (July, 2011), 5–8