Intensity Particle Flow For SMC-PHD Filter

Yang Liu, Shuguo Yang, Volkan Kılıç, Yunpeng Li, Yuhui Luo, Adrian Hilton, Wenwu Wang

***Abstract*—Non-zero diffusion particle flow (NPF) has emerged as a useful tool to mitigate the weight degeneracy problem in sequential Monte Carlo based probability hypothesis density (SMC-PHD) filter for multi-target tracking. However, NPF is calculated for each particle with only a single measurement, resulting in running a data association algorithm between measurements and particles. When the target disappears from the scene, the particles may be associated with other targets or clutters leading to degradation in tracking performance. To address these limitations, we propose an intensity particle flow (IPF) algorithm based on the clutter intensity and detection probability without using data association. By employing a novel birth model, the computational cost of the proposed method can be further reduced. Experimental evaluations on simulated data and real data such as the AV16.3 and LOCATA datasets show the significantly improved tracking accuracy by our proposed filter as compared with several baseline methods for multi-target tracking.**

***Index Terms*—Tracking system, SMC-PHD, particle flow, multi-target.**

I. Introduction

MULTI-TARGET tracking in an enclosed space is an important problem in several subject areas such as smart surveillance systems [1], advanced human-computer interfaces [2], sports video analysis [3], target discrimination [4], and audio-visual speech recognition [5]. Bayesian inference framework has provided an intuitive way for the estimation of target states in a dynamic system [6]. However, tractable solutions to the recursive Bayesian estimation problem are only available for the linear Gaussian model or the finite state Hidden Markov model (HMM). For nonlinear state estimation, extended Kalman filter (EKF) [7] and unscented Kalman filter (UKF) [8] are often used. Particle filter [9], as an alternative method, can estimate the target state by a weighted set of random particles. However, it works with the assumption that the number of targets is known and invariant.

To track an unknown and time-varying number of targets with clutter, mis-detection and data association uncertainty, a number of filters have been introduced, such as the joint probabilistic data association filter [10], multiple hypothesis tracking [11], and random finite set (RFS) [12], [13]. The RFS has a good performance in the presence of false detection, detection uncertainty, and mis-detection, however, its computational complexity grows significantly with the increase in the number of targets.

To solve this problem, the multi-object Bayesian recursion is approximated by propagating the probability hypothesis density (PHD) [12], [14], based on the first order moment of an RFS, with several well-known implementations, such as the sequential Monte Carlo (SMC) PHD (SMC-PHD) filter [15], the Gaussian mixture PHD (GM-PHD) filter [16] and the cardinalized PHD (CPHD) filter [17]. The GM-PHD filter provides a closed form solution under the linear Gaussian model [16], while the SMC-PHD filter is designed for the non-Gaussian model. The SMC-PHD filter offers generally better performance than GM-PHD, but suffers from the weight degeneracy problem, i.e. only a small number of particles retain significant weights and the effective sample size (ESS) becomes small after a number of iterations [18].

Several methods have been developed to address the weight degeneracy problem. In the SMC-based methods e.g. the auxiliary particle filter [19], the unscented particle filter [20], auxiliary SMC-PHD filter [21], and the unscented auxiliary cardinalized PHD filter [22], the variance of the importance weights is minimized by approximating the optimal proposal distribution with the most recent measurements, leading to an increase in ESS. Alternatively, the particles are independently sampled to represent the target posterior, as performed in the Markov Chain Monte Carlo method [23]. Another idea is to approach the true posterior density from the prior density by using bridging densities [24], [25], which was shown to give promising performance, by approximating the optimal densities.

Recently, particle flow has been proposed [26], [27], [28], which migrates particles from the prior distribution to the posterior distribution based on a homotopy function [29]. In terms of how the homotopy function is solved, the particle flow methods can be divided into six categories including incompressible particle flow [30], zero diffusion particle flow (ZPF) [28], Coulomb's law particle flow [31], zero-curvature particle flow (ZPF) [32], non-zero diffusion particle flow (NPF) [33] and stochastic particle flow with Gromov's method [34]. With the assumption that the prior and likelihood follow Gaussian distributions, special types of ZPF have been derived, such as the exact Daum and Huang (EDH) filter [29], the localized exact Daum and Huang (LEDH) filter [35] and the Gaussian particle flow importance sampling (GPFIS) filter [28], and they have been used in [35], [36], [37], [38].

The particle flow filters have been used to improve the

Y. Liu, A Hilton and W. Wang are with the Centre for Vision, Speech and Signal Processing, Department of Electrical and Electronic Engineering, University of Surrey, Guildford, GU2 7XH, U.K. E-mails: [yangliu; w.wang; a.hilton]@surrey.ac.uk.

S. Yang is with the School of Mathematics and Physics, Qingdao University of Science and Technology, Qingdao, China. E-mail: ysg_2005@163.com.

V. Kılıç is with Department of Electrical and Electronics Engineering, Izmir Katip Celebi University, 35620 Cigli-Izmir, Turkey. E-mail: volkan.kilic@ikc.edu.tr.

Y. Li is with Department of Computer Science, University of Surrey, Guildford, GU2 7XH, U.K. E-mail: yunpeng.li@surrey.ac.uk.

Y. Luo is with Department of Data Science, National Physical Laboratory, Teddington, TW11 0LW, U.K. E-mail: yuhui.luo@npl.co.uk.

This work was supported by the EPSRC Programme Grant S3A: Future Spatial Audio for an Immersive Listener Experience at Home (EP/L000539/1), the BBC as part of the BBC Audio Research Partnership and the China Scholarship Council (CSC).

PHD filters, including the SMC-PHD filters, as in ZPF-SMC-PHD [39], [40], [41], NPF-SMC-PHD [42], and the GM-PHD filters, as in Gaussian particle flow PHD (GPF-PHD) [43] and Gaussian mixture particle flow PHD [44]. Based on Bayes recursion, the particle flow often uses only a single measurement to update the particles, as a result, data association between the measurements and the particles is required. By duplicating the particles, particle flow is independently calculated for each individual measurement and particle, which, however, induces a high computational cost. Moreover, in the ZPF-SMC-PHD [39], [40], [41], NPF-SMC-PHD [42], GPF-PHD [43] and δ-GLMB particle [45] filters, the clutter density and detection probability have been used to update the weights of the particles, but not their states, and this may degrade the tracking performance.

In this paper, we propose a new filter called intensity particle flow SMC-PHD filter (IPF-SMC-PHD), where the intensity function of the clutter and the probability of target detection are used to update both the states and the weights of the particles. More specifically, the standard Bayes recursion of NPF is replaced by the intensity recursion associated with the multi-target posterior. Based on the differentiation of the intensity function with respect to the synthetic time and particle state, the intensity particle flow is calculated by the intensity function of the clutter, the probability of target detection and multi-target likelihood. Compared with ZPF and NPF, IPF can mitigate the clutters and mis-detection problem via replacing the data association based likelihood function by the multi-target likelihood. We demonstrate the performance of the IPF-SMC-PHD filter on simulated data and real data, such as the LOCATA dataset and AV16.3 dataset, which show the significantly improved tracking accuracy by our proposed filter as compared with several baseline methods for multi-target tracking with clutters, mis-detection and data association uncertainty.

The remainder of the paper is organized as follows: the next section presents the problems and related works. Section III introduces the details of the proposed IPF. Section IV describes the implementation of the proposed IPF-SMC-PHD filter. Section V details the experimental setup and results. Concluding remarks are given in Section VI.

II. PROBLEM STATEMENT AND BACKGROUND

This section describes our problem formulation, the NPF-SMC-PHD filter and the intensity function. For multi-target tracking, we assume that the target dynamics and measurements are described as:

$$\{\tilde{\boldsymbol{m}}_k^j\}_{j=1}^{\tilde{N}_k} = \mathbf{F}_{\tilde{\boldsymbol{m}}}\left(\{\tilde{\boldsymbol{m}}_{k-1}^j\}_{j=1}^{\tilde{N}_{k-1}}, \boldsymbol{\tau}_k\right) \tag{1}$$

$$\boldsymbol{Z}_k = \mathbf{F}_{\boldsymbol{z}}\left(\{\tilde{\boldsymbol{m}}_k^j\}_{j=1}^{\tilde{N}_k}, \boldsymbol{\varsigma}_k\right) + \epsilon_k \tag{2}$$

where $\tilde{\boldsymbol{m}}_k^j \in \mathbb{R}^M$ represents the state vector for the jth target at time k, $\tilde{}$ is used to distinguish the target state from the particle state used later, and $\tilde{N}_k$ is the number of targets at time k. Let $\boldsymbol{Z}_k$ denote the set of measurements at time k defined as $\{\boldsymbol{z}_k^r\}_{r=1}^{R_k}$ where R_k is the number of measurements. The state $\tilde{\boldsymbol{m}}_k^j = [x_k^j, y_k^j, \dot{x}_k^j, \dot{y}_k^j]^T$ consists of positions (x_k^j, y_k^j) and velocities $(\dot{x}_k^j, \dot{y}_k^j)$, while the observation is a noisy version of the position. Hence $M = 4$. For 3D calculations, the target state is set as $\tilde{\boldsymbol{m}}_k^j = [x_k^j, y_k^j, w_k^j, \dot{x}_k^j, \dot{y}_k^j, \dot{w}_k^j]^T$, where (x_k^j, y_k^j, w_k^j) is the 3D position of the jth target and $(\dot{x}_k^j, \dot{y}_k^j, \dot{w}_k^j)$ is the target velocity. $\boldsymbol{\tau}_k$ and $\boldsymbol{\varsigma}_k$ are system excitation and measurement noise terms, respectively. ϵ_k is the clutter term. $\mathbf{F}_{\tilde{\boldsymbol{m}}}$ represents the state transition model and $\mathbf{F}_{\boldsymbol{z}}$ is the nonlinear measurement model.

A. Intensity function

Intensity function, also called PHD, is the first-order moment of the multi-object posterior. Since the intensity function is associated with the space where individual targets live, its propagation requires less computational load than the multi-target Bayes filter [46]. The update step of the intensity function is calculated as:

$$\Psi_k^i = \left[1 - p_{D,k}^i + \sum_{r=1}^{R_k} \frac{p_{D,k}^i h_k^{i,r}}{\kappa_k + \sum_{j=1}^{N_k} p_{D,k}^i h_k^{i,r} \Psi_{k|k-1}^i}\right] \Psi_{k|k-1}^i \tag{3}$$

where $p_{D,k}^i$ is the detection probability that the measurements $\boldsymbol{Z}_k$ are collected from the particle $\boldsymbol{m}_k^i$, and κ_k is the intensity function of clutter at time k. $h_k^{i,r}$ is the multi-target likelihood of the ith particle associated with the rth measurement at time k. Ψ_k^i is the abbreviation of the intensity function $\Psi_k^i(\boldsymbol{m}_k^i|\boldsymbol{Z}_k) = \int \delta(\boldsymbol{m}_k^i)\psi_k^i(\boldsymbol{M}_k|\boldsymbol{Z}_k)\delta\boldsymbol{M}$ where $\psi_k^i(\boldsymbol{M}_k|\boldsymbol{Z}_k)$ is the posterior density of the ith particle at time k, denoted as ψ_k^i for convenience.

Compared to the Bayesian density function used by the NPF, the intensity function considers more target information, such as the detection probability $p_{D,k}^i$, the intensity function of clutter κ_k and the multi-target likelihood $h_k^{i,r}$. However, the direct application of particle flow methods to propagate the intensity function would fail because the intensity function is not a probability density function. In the SMC-PHD filter [15], Ψ_k^i is implemented by the particle weight, which is discussed later in Eq. (17). In our proposed IPF, Ψ_k^i is calculated by the covariance matrix of the ith particle, which is discussed later in Section III-C.

B. NPF-SMC-PHD filter

In [42], the NPF-SMC-PHD filter (i.e. Algorithm 1) is proposed for multi-target tracking. At time $k-1$, the target PHD is approximated by N_{k-1} particles $\{\boldsymbol{m}_{k-1}^i\}_{i=1}^{N_{k-1}}$ and their weights $\{\omega_{k-1}^i\}_{i=1}^{N_{k-1}}$, where N_{k-1} is the number of particles at time $k-1$. In the prediction step [46], the particle set is obtained by the proposal distribution $q_k(\boldsymbol{m}_{k|k-1}^i|\boldsymbol{m}_{k-1}^i, \boldsymbol{Z}_k)$,

$$\boldsymbol{m}_{k|k-1}^i \propto q_k(\cdot|\boldsymbol{m}_{k-1}^i, \boldsymbol{Z}_k) \tag{4}$$

Their weights are

$$\omega_{k|k-1}^i = \frac{p_s \phi_k(\boldsymbol{m}_{k|k-1}^i|\boldsymbol{m}_{k-1}^i)\omega_{k-1}^i}{q_k(\boldsymbol{m}_{k|k-1}^i|\boldsymbol{m}_{k-1}^i, \boldsymbol{Z}_k)} \quad i = 1, ..., N_{k-1} \tag{5}$$

where $\phi_k(\boldsymbol{m}_{k|k-1}^i|\boldsymbol{m}_{k-1}^i)$ is the transition distribution and p_s is the survival distribution. Since the proposal distribution is

normally assumed to be the same as the transition distribution [46], [47], [39], Eq. (5) can be simplified as:

$$\omega_{k|k-1}^{i} = p_s \omega_{k-1}^{i} \tag{6}$$

In addition, N_B particles are sampled from the new born importance function p_k,

$$\boldsymbol{m}_{k|k-1}^{i} \propto p_k(\cdot|\boldsymbol{Z}_k) \tag{7}$$

Their weights are

$$\omega_{k|k-1}^{i} = \frac{\gamma_k(\boldsymbol{m}_{k|k-1}^{i})}{N_B p_k(\boldsymbol{m}_{k|k-1}^{i}|\boldsymbol{Z}_k)} \quad i = N_{k-1}+1, ..., N_{k-1}+N_B \tag{8}$$

where γ_k is the probability of new born targets, whose integral approximates the average number of targets in the state space. N_{k-1} is the number of surviving particles at time $k-1$.

After predicting the particles, a particle flow migrates the particle states via the Ito stochastic differential equation [29] and can be approximated using an Euler integration scheme:

$$\triangle \boldsymbol{m}_{k|k-1}^{i} = \boldsymbol{f}_k^{i}(\boldsymbol{m}_{k|k-1}^{i}, \lambda)\triangle\lambda + \upsilon_k^{i}\boldsymbol{w}_k^{i} \tag{9}$$

where $\boldsymbol{f}_k^{i} \in \mathbb{R}^M$ is the particle flow vector which moves the particle $\boldsymbol{m}_{k|k-1}^{i}$ with the distance $\triangle\boldsymbol{m}_{k|k-1}^{i}$ at λ, $\boldsymbol{w}_k^{i} \in \mathbb{R}^M$ is the Wiener process with the diffusion coefficient υ_k^{i}, and λ, called the synthetic time, is the step size parameter taking values from $[0, \triangle\lambda, 2\triangle\lambda, \cdots, N_\lambda\triangle\lambda]$ with $N_\lambda\triangle\lambda = 1$. In NPF [33], $\boldsymbol{f}_k^{i} \in \mathbb{R}^M$ is given by,

$$\boldsymbol{f}_k^{i} = -[\boldsymbol{\nabla}^2 \ln \psi_k^{i}]^{-1}(\boldsymbol{\nabla} \ln h_k^{i}) \tag{10}$$

where

$$\boldsymbol{\nabla}^2 \ln \psi_k^{i} \approx -(\boldsymbol{P}_{k-1}^{i})^{-1} + \lambda \boldsymbol{\nabla}^2 \ln h_k^{i} \tag{11}$$

where $\boldsymbol{P}_{k-1}^{i}$ is the covariance matrix of the ith particle at time $k-1$, $\boldsymbol{\nabla}$ is the spatial vector differentiation operator $\frac{\partial}{\partial \boldsymbol{m}_{k|k-1}^{i}}$, and the likelihood h_k^{i} is calculated by the data association function,

$$h_k^{i} = \min_r h_k^{i,r} \tag{12}$$

With a Gaussian model $h_k^{i,r}$, the likelihood h_k^{i} can be written as

$$h_k^{i} = \mathcal{N}(\boldsymbol{m}_{k|k-1}^{i} | \operatorname*{argmin}_{\boldsymbol{z}_k^r} \left\| \boldsymbol{z}_k^r - \boldsymbol{m}_{k|k-1}^{i} \right\|_2, \boldsymbol{R}) \tag{13}$$

where $\boldsymbol{R}$ is the covariance matrix of the measurement noise, and $\|\cdot\|_2$ is the l_2 norm. Then each particle state is updated as

$$\boldsymbol{m}_{k|k-1}^{i} \Leftarrow \boldsymbol{m}_{k|k-1}^{i} + \triangle\boldsymbol{m}_{k|k-1}^{i} \tag{14}$$

For the invertible particle flow, the weights are adjusted by [48]:

$$\omega_{k|k-1}^{i} \Leftarrow \frac{q_k(\boldsymbol{m}_{k|k-1}^{i}|\tilde{\boldsymbol{m}}_{k|k-1}^{j}) \left|\det(\boldsymbol{G}_k^{i})\right|}{q_k(\boldsymbol{m}_{k|k-1}^{i} + \triangle\boldsymbol{m}_{k|k-1}^{i}|\tilde{\boldsymbol{m}}_{k|k-1}^{j})}\omega_{k|k-1}^{i} \tag{15}$$

where

$$\boldsymbol{G}_k^{i} = \prod_{\lambda=1}^{N_\lambda}(\boldsymbol{I} + \Delta\lambda\boldsymbol{\nabla}\boldsymbol{f}_{k,\lambda}^{i}) \tag{16}$$

where $\det(\cdot)$ denotes determinant and $|\cdot|$ denotes the absolute value. The proposal distribution $q_k(\boldsymbol{m}_{k|k-1}^{i}|\tilde{\boldsymbol{m}}_{k|k-1}^{j}) \propto \mathcal{N}(\tilde{\boldsymbol{m}}_{k|k-1}^{j}, \boldsymbol{\Sigma}_q)$. Finally, the weights are calculated by the PHD filter as

$$\omega_k^{i} = \left[1 - p_{D,k}^{i} + \sum_{r=1}^{R_k} \frac{p_{D,k}^{i} h_k^{i}}{\kappa_k + \sum_{i=1}^{N_k} p_{D,k}^{i} h_k^{i,r} \omega_{k|k-1}^{i}}\right] \omega_{k|k-1}^{i} \tag{17}$$

The number of targets is estimated as the sum of the weights,

$$\tilde{N}_k = \sum_{i=1}^{N_k} \omega_k^{i} \tag{18}$$

The states and weights of the targets $\{\tilde{\boldsymbol{m}}_k^{j}, \tilde{\omega}_k^{j}\}_{j=1}^{\tilde{N}_k}$ can be calculated using e.g. K-means clustering method [49] or multi-expected a posterior (MEAP) [47], [50]. Finally, resampling is performed when the ESS [51] is smaller than half of the number of particles. In the resampling step, we can obtain $\{\boldsymbol{m}_k^{i}, \omega_k^{i}\}_{i=1}^{N_k}$, where $\{\omega_k^{i}\}_{i=1}^{N_k} = 1/N_k$.

Algorithm 1 NPF-SMC-PHD Filter

Input: $\{\boldsymbol{m}_{k-1}^{i}, \omega_{k-1}^{i}\}_{i=1}^{N_{k-1}}$, $\{\boldsymbol{P}_{k-1}^{i}\}_{i=1}^{N_{k-1}}$, k and $\boldsymbol{Z}_k$.
Output: $\{\tilde{\boldsymbol{m}}_k^{j}, \tilde{\omega}_k^{j}\}_{j=1}^{\tilde{N}_k}$, $\{\boldsymbol{P}_k^{i}\}_{i=1}^{N_k}$ and $\{\boldsymbol{m}_k^{i}, \omega_k^{i}\}_{i=1}^{N_k}$.
Initialize: q_k, p_s, ϕ_k, p_k, κ_k, $P_{D,k}$, $\triangle\lambda$, N_λ, υ_k^{i}, $\boldsymbol{w}_k^{i}$, $\boldsymbol{R}$ and N_B.
Run:
Propagate the particle states $\{\boldsymbol{m}_{k|k-1}^{i}\}_{i=1}^{N_{k-1}}$ by Eq. (4).
Calculate the particle weights $\{\omega_{k|k-1}^{i}\}_{i=1}^{N_{k-1}}$ by Eq. (6).
Sample N_B new born particles $\{\boldsymbol{m}_{k|k-1}^{i}, \omega_{k|k-1}^{i}\}_{i=N_{k-1}+1}^{N_{k-1}+N_B}$ uniformly around each measurement by Eq. (7) and Eq. (8).
Combine all the particles: $\{\boldsymbol{m}_{k|k-1}^{i}, \omega_{k|k-1}^{i}\}_{i=1}^{N_k} = \{\boldsymbol{m}_{k|k-1}^{i}, \omega_{k|k-1}^{i}\}_{i=1}^{N_{k-1}} \cup \{\boldsymbol{m}_{k|k-1}^{i}, \omega_{k|k-1}^{i}\}_{i=N_{k-1}+1}^{N_{k-1}+N_B}$.
for $\lambda \in [0, \triangle\lambda, 2\triangle\lambda, \cdots, N_\lambda\triangle\lambda]$ **do**
  Calculate the likelihood h_k^{i} by Eq. (13).
  Calculate particle flow $\boldsymbol{f}_k^{i}$ by Eq. (10).
  Calculate $\triangle\boldsymbol{m}_{k|k-1}^{i}$ by Eq. (9).
  Update each particle weight by Eq. (15).
  Update each particle state by Eq. (14).
end for
Update the particle weights $\{\omega_{k|k-1}^{i}\}_{i=1}^{N_k}$ to obtain $\{\omega_k^{i}\}_{i=1}^{N_k}$ by Eq. (17) and calculate $\tilde{N}_k = \sum_{i=1}^{N_k} \omega_k^{i}$.
Set $\{\boldsymbol{m}_k^{i}\}_{i=1}^{\tilde{N}_k}$ as $\{\boldsymbol{m}_{k|k-1}^{i}\}_{i=1}^{N_k}$.
Cluster particles and get $\{\tilde{\boldsymbol{m}}_k^{j}, \tilde{\omega}_k^{j}\}_{j=1}^{\tilde{N}_k}$ by the K-means or the MEAP method.
Calculate $\{\boldsymbol{P}_k^{i}\}_{i=1}^{N_k}$ by e.g. a Kalman filter.
if ESS $< N_k/2$ **then**
  Resample $\{\boldsymbol{m}_k^{i}, \omega_k^{i}\}_{i=1}^{N_k}$.
end if

The NPF can mitigate the weight degeneracy problem in the SMC-PHD filter. However, the NPF has two limitations. The NPF assumes that all the measurements are generated by targets and all targets stay in the scene. Although $p_{D,k}^{i}$ and κ_k are used to update the particle weights, they are not considered for the update of the particle states in the particle flow. Apart from that, the NPF considers the measurements in the neighborhood of the particle determined by a data association step [42]. However, data association is hard to be

achieved accurately, especially when the particles are located in the middle of several measurements.

III. Intensity particle flow

To address the limitations of NPF, we propose to exploit both the clutter density and the detection probability in terms of the intensity function. First, we calculate the intensity function based on the log-homotopy function. Then, the differentiation of the intensity functions is calculated with respect to the synthetic time and the particle state, respectively. Finally, the equations and complexity analysis of IPF are given.

A. Intensity function with the log-homotopy

Based on the log-homotopy function, we can model the motion of the particles from the prior to the posterior densities, similar to the idea used in ZPF [30]. We extend this idea for the calculation of the intensity function of the SMC-PHD filter. For each particle, we take the natural logarithm of both sides of Eq. (3),

$$\begin{aligned}\ln(\Psi_k^i) &= \ln(\left(1-p_{D,k}^i+p_{D,k}^i C_k^i\right)\Psi_{k|k-1}^i)\\ &= \ln(1-p_{D,k}^i+p_{D,k}^i C_k^i)+\ln(\Psi_{k|k-1}^i),\end{aligned} \tag{19}$$

where

$$C_k^i = \sum_{r=1}^{R_k}\frac{h_k^{i,r}}{\kappa_k+\sum_{i=1}^{N_k} p_{D,k}^i h_k^{i,r}\Psi_{k|k-1}^i}. \tag{20}$$

Based on the log-homotopy function which relates the prior to the posterior intensity function, Eq. (19) becomes

$$\ln(\Psi_{k,\lambda}^i) = \lambda\ln(1-p_{D,k}^i+p_{D,k}^i C_k^i)+\ln(\Psi_{k|k-1}^i), \tag{21}$$

When $\lambda = 0$, $\Psi_{k,\lambda}^i$ represents the prior intensity function $\Psi_{k|k-1}^i$. When λ is varied to 1, $\Psi_{k,\lambda}^i$ is translated into the normalized posterior intensity function Ψ_k^i.

B. Differentiation of the intensity function on the synthetic time

Since the Fokker-Planck equation [52] relates the motion of a particle with the evolution of the density for its position, it is used to calculate the particle flow $\boldsymbol{f}_k^i$,

$$\frac{\partial\Psi_{k,\lambda}^i}{\partial\lambda} = \frac{\mathrm{div}[\mathbf{Q}_k^i\boldsymbol{\nabla}\Psi_{k,\lambda}^i]}{2} - \mathrm{div}\left(\boldsymbol{f}_k^i\Psi_{k,\lambda}^i\right), \tag{22}$$

where $\mathbf{Q}_k^i$ is the covariance matrix of the process noise, and div is the divergence operator [53], e.g. $\mathrm{div}(\boldsymbol{f}_k^i) = \sum_{l=1}^{M}\frac{\partial \boldsymbol{f}_k^i}{\partial \boldsymbol{\varepsilon}^l}$ where $\boldsymbol{\varepsilon}^l \in \mathbb{R}^M$ is the lth basis vector. $\boldsymbol{\nabla}$ is the spatial vector differentiation operator $\frac{\partial}{\partial \boldsymbol{m}_{k|k-1}^i}$. The differentiation of $\Psi_{k,\lambda}^i$ with respect to λ is given as $\frac{\partial\Psi_{k,\lambda}^i}{\partial\lambda}$. Differentiating $\Psi_{k,\lambda}^i$ with respect to λ based on Eq. (21) becomes

$$\frac{\partial\Psi_{k,\lambda}^i}{\partial\lambda} = \Psi_{k,\lambda}^i\ln(1+p_{D,k}^i C_k^i - p_{D,k}^i). \tag{23}$$

Based on Eq. (22) and Eq. (23), we can get:

$$\ln(1+p_{D,k}^i C_k^i - p_{D,k}^i) = \frac{\mathrm{div}[\mathbf{Q}_k^i\boldsymbol{\nabla}\Psi_{k,\lambda}^i]}{2\Psi_{k,\lambda}^i} - \frac{\mathrm{div}\left(\boldsymbol{f}_k^i\Psi_{k,\lambda}^i\right)}{\Psi_{k,\lambda}^i}. \tag{24}$$

Computing the divergence of $\boldsymbol{f}_k^i\Psi_{k,\lambda}^i$ using the calculus as in [33] leads to:

$$\mathrm{div}\left(\boldsymbol{f}_k^i\Psi_{k,\lambda}^i\right) = \Psi_{k,\lambda}^i(\mathrm{div}(\boldsymbol{f}_k^i)+\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i)\boldsymbol{f}_k^i). \tag{25}$$

As a result, Eq. (24) becomes

$$\begin{aligned}&\ln(-p_{D,k}^i+p_{D,k}^i C_k^i+1)\\ =&\frac{\mathrm{div}[\mathbf{Q}_k^i\boldsymbol{\nabla}\Psi_{k,\lambda}^i]}{2\Psi_{k,\lambda}^i} - \mathrm{div}(\boldsymbol{f}_k^i) - \boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i)\boldsymbol{f}_k^i,\end{aligned} \tag{26}$$

Differentiating Eq. (26) with respect to $\boldsymbol{m}_{k|k-1}^i$, we have

$$\begin{aligned}&\boldsymbol{\nabla}\ln(-p_{D,k}^i+p_{D,k}^i C_k^i+1)+\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i))\boldsymbol{f}_k^i\\ &= \boldsymbol{\nabla}(\frac{\mathrm{div}[\mathbf{Q}_k^i\boldsymbol{\nabla}\Psi_{k,\lambda}^i]}{2\Psi_{k,\lambda}^i}) - \boldsymbol{\nabla}(\mathrm{div}(\boldsymbol{f}_k^i)),\end{aligned} \tag{27}$$

where $\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i))$ is the second differentiation of $\ln(\Psi_{k,\lambda}^i)$ with respect to $\boldsymbol{m}_{k|k-1}^i$,

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i))\boldsymbol{f}_k^i = \boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i)\boldsymbol{\nabla}(\boldsymbol{f}_{k,\lambda}^i)+\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i))\boldsymbol{f}_k^i. \tag{28}$$

Based on the non-zero particle flow [33], [42], there is a diffusion matrix $\boldsymbol{Q}_k^i$ such that

$$\boldsymbol{\nabla}(\frac{\mathrm{div}[\mathbf{Q}_k^i\boldsymbol{\nabla}\Psi_{k,\lambda}^i]}{2\Psi_{k,\lambda}^i}) = \boldsymbol{\nabla}(\mathrm{div}(\boldsymbol{f}_k^i)) + \boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i)\boldsymbol{\nabla}(\boldsymbol{f}_{k,\lambda}^i). \tag{29}$$

Then, Eq. (27) is simplified as:

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}(\ln\Psi_{k,\lambda}^i))\boldsymbol{f}_{k,\lambda}^i = -\boldsymbol{\nabla}\ln(-p_{D,k}^i+p_{D,k}^i C_k^i+1). \tag{30}$$

If we assume $\boldsymbol{\nabla}(\boldsymbol{\nabla}(\ln\Psi_{k,\lambda}^i))$ is invertible, we can get

$$\boldsymbol{f}_{k,\lambda}^i = -(\boldsymbol{\nabla}(\boldsymbol{\nabla}(\ln\Psi_{k,\lambda}^i)))^{-1}\boldsymbol{B}_k^i, \tag{31}$$

where

$$\begin{aligned}\boldsymbol{B}_k^i =&\boldsymbol{\nabla}\ln(-p_{D,k}^i+p_{D,k}^i C_k^i+1)\\ =&\frac{\boldsymbol{\nabla}p_{D,k}^i(C_k^i-1)+\boldsymbol{\nabla}C_k^i p_{D,k}^i}{1-p_{D,k}^i+p_{D,k}^i C_k^i},\end{aligned} \tag{32}$$

where C_k^i is a scalar, $\boldsymbol{f}_{k,\lambda}^i \in \mathbb{R}^{M\times 1}$ and $\boldsymbol{B}_k^i \in \mathbb{R}^{M\times 1}$. As $p_{D,k}^i$ is normally set as a constant $p_{D,k}$ for different particles [47], [15] and $\boldsymbol{\nabla}p_{D,k}^i = 0$, we can get

$$\boldsymbol{B}_k^i = \frac{\boldsymbol{\nabla}C_k^i p_{D,k}}{1-p_{D,k}+p_{D,k}C_k^i}. \tag{33}$$

C. Differentiation of the intensity function on the particle state

Based on Eq. (21), the particle flow can be calculated by the differentiation of the intensity function on the particle state. Differentiating $\ln(\Psi_{k,\lambda}^i)$ with respect to $\boldsymbol{m}_{k|k-1}^i$, we can get

$$\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i) = \lambda\boldsymbol{B}_k^i + \boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i), \tag{34}$$

and

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k,\lambda}^i)) = \lambda\boldsymbol{\nabla}(\boldsymbol{B}_k^i) + \boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i)). \tag{35}$$

Based on (35), (31) becomes

$$\boldsymbol{f}_{k,\lambda}^i = -(\lambda\boldsymbol{\nabla}(\boldsymbol{B}_k^i) + \boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i)))^{-1}\boldsymbol{B}_k^i, \tag{36}$$

where

$$\begin{aligned}&\boldsymbol{\nabla}(\boldsymbol{B}_k^i)\\=&\frac{p_{D,k}^i(1-p_{D,k}^i+p_{D,k}^iC_k^i)\boldsymbol{\nabla}^2C_k^i-(p_{D,k}^i)^2\boldsymbol{\nabla}C_k^i(\boldsymbol{\nabla}C_k^i)^T}{(1-p_{D,k}^i+p_{D,k}^iC_k^i)^2},\end{aligned} \tag{37}$$

$$\begin{aligned}\boldsymbol{\nabla}C_k^i=&\boldsymbol{\nabla}\sum_{r=1}^{R_k}\frac{h_k^{i,r}}{\kappa_k+\sum_{i=1}^{N_k}p_{D,k}^ih_k^{i,r}\Psi_{k|k-1}^i}\\=&\sum_{r=1}^{R_k}\frac{\boldsymbol{\nabla}h_k^{i,r}}{\kappa_k+\sum_{i=1}^{N_k}p_{D,k}^ih_k^{i,r}\Psi_{k|k-1}^i},\end{aligned} \tag{38}$$

and

$$\boldsymbol{\nabla}^2C_k^i=\sum_{r=1}^{R_k}\frac{\boldsymbol{\nabla}(\boldsymbol{\nabla}h_k^{i,r})}{\kappa_k+\sum_{i=1}^{N_k}p_{D,k}^ih_k^{i,r}\Psi_{k|k-1}^i}, \tag{39}$$

where $\kappa_k+\sum_{i=1}^{N_k}p_{D,k}^ih_k^{i,r}\Psi_{k|k-1}^i$ is a normalizing constant for particles. As shown in Eq. (36), the particle flow procedure requires $\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i))$. If we assume the particle state intensity function is Gaussian, we can calculate $\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i))$ as

$$\begin{aligned}&\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i))\\=&\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\frac{e^{-\frac{1}{2}(\boldsymbol{m}_{k|k-1}^i-\overline{\boldsymbol{m}}_{k|k-1}^i)^T(\boldsymbol{P}_{k|k-1}^i)^{-1}(\boldsymbol{m}_{k|k-1}^i-\overline{\boldsymbol{m}}_{k|k-1}^i)}}{\sqrt{(2\pi)^M|\boldsymbol{P}_{k|k-1}^i|}}))\\=&\boldsymbol{\nabla}(\boldsymbol{\nabla}(-\frac{1}{2}(\boldsymbol{m}_{k|k-1}^i-\overline{\boldsymbol{m}}_{k|k-1}^i)^T(\boldsymbol{P}_{k|k-1}^i)^{-1}\\&(\boldsymbol{m}_{k|k-1}^i-\overline{\boldsymbol{m}}_{k|k-1}^i)-\ln\sqrt{(2\pi)^M|\boldsymbol{P}_{k|k-1}^i|}))\\=&-\frac{1}{2}\boldsymbol{\nabla}(((\boldsymbol{P}_{k|k-1}^i)^{-1}+((\boldsymbol{P}_{k|k-1}^i)^{-1})^T)(\boldsymbol{m}_{k|k-1}^i-\overline{\boldsymbol{m}}_{k|k-1}^i))\\=&-\frac{1}{2}((\boldsymbol{P}_{k|k-1}^i)^{-1}+((\boldsymbol{P}_{k|k-1}^i)^{-1})^T),\end{aligned} \tag{40}$$

where M is dimensionality of the particle state, $\overline{\boldsymbol{m}}_{k|k-1}^i$ is the mean value of $\boldsymbol{m}_{k|k-1}^i$ and $\boldsymbol{P}_{k|k-1}^i$ is the covariance matrix. As $\boldsymbol{P}_{k|k-1}^i$ is a symmetric matrix, $(\boldsymbol{P}_{k|k-1}^i)^{-1}=((\boldsymbol{P}_{k|k-1}^i)^{-1})^T$. Therefore, $\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i))$ is given by:

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}\ln(\Psi_{k|k-1}^i))=-(\boldsymbol{P}_{k|k-1}^i)^{-1}, \tag{41}$$

where the covariance matrix $\boldsymbol{P}_{k|k-1}^i$ of the prediction error for the prior distribution can be estimated by the Kalman filter or by a clustering method. $\boldsymbol{P}_{k|k-1}^i$ is an invertible matrix since the dimensions of the particle state are independent. For the Kalman filter, $\boldsymbol{P}_{k|k-1}^i$ is independently updated for each particle. When the dynamic model is not linear, the extended Kalman filter or the unscented Kalman filter [54] can be used to estimate $\boldsymbol{P}_{k|k-1}^i$, where the computational complexity is $\mathcal{O}(M^3)$ [55]. For the clustering method, we can choose, e.g. the k-means method, to cluster the particles, and estimate the mean and covariance matrix for each cluster. Its computational complexity is only $\mathcal{O}(M^2)$.

The computational complexity of the IPF is linear with respect to the number of measurements, the number of particles and the number of flow update steps N_λ. The most computationally intensive part of IPF is Eq. (33), which has a complexity of $\mathcal{O}(M^3)$. The computational complexity of the update step of IPF is the same as that of the ZPF and NPF.

IV. IPF-SMC-PHD FILTER

In this section, we use the IPF to improve the NPF-SMC-PHD filter considering the fact that the diffusion matrices of NPF and IPF are the same. The IPF can be used to replace the NPF in the state update step, before updating the weights.

A. IPF in IPF-SMC-PHD filter

The IPF is used to update the states of the particles, in particular, the surviving particles, after the prediction step and before the update step. However, the born particles are created based on the measurements, therefore, they will not be adjusted by the particle flow.

After applying Eq. (4) and Eq. (5), the particle set is obtained as $\{\boldsymbol{m}_{k|k-1}^i,\omega_{k|k-1}^i\}_{i=1}^{N_{k-1}}$. In the NPF-SMC-PHD filter, the particle flow is calculated according to

$$\boldsymbol{f}_{k,\lambda}^i=-(\lambda\boldsymbol{\nabla}(\boldsymbol{B}_k^i)-(\boldsymbol{P}_{k|k-1}^i)^{-1})^{-1}\boldsymbol{B}_k^i, \tag{42}$$

and

$$C_k^i=\sum_{r=1}^{R_k}\frac{h_k^{i,r}}{\kappa_k+\sum_{i=N_{k-1}+1}^{N_{k-1}+N_B}S_k^{i,r}+\sum_{i=1}^{N_{k-1}}h_k^{i,r}\omega_{k|k-1}^i}, \tag{43}$$

where $S_k^{i,r}$ is the birth intensity function for the ith particle and the rth measurement at k. With a Gaussian likelihood model, the particle flow Eq. (42) for particle motion may be derived analytically. The differentiation of the likelihood $h_k^{i,r}$ with respect to $\boldsymbol{m}_{k|k-1}^i$ is calculated as follows:

$$\boldsymbol{\nabla}h_k^{i,r}=-h_k^{i,r}\boldsymbol{R}^{-1}(\boldsymbol{m}_{k|k-1}^i-\boldsymbol{z}_k^r), \tag{44}$$

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}h_k^{i,r})=h_k^{i,r}(\boldsymbol{D}_k^{i,r}-\boldsymbol{R}^{-1}), \tag{45}$$

$$\boldsymbol{D}_k^{i,r}=\boldsymbol{R}^{-1}(\boldsymbol{m}_{k|k-1}^i-\boldsymbol{z}_k^r)(\boldsymbol{m}_{k|k-1}^i-\boldsymbol{z}_k^r)^T(\boldsymbol{R}^{-1})^T. \tag{46}$$

With the increment of λ, the change rate of $\triangle\boldsymbol{m}_{k|k-1}^i$ decreases. If $\left\|\triangle\boldsymbol{m}_{k|k-1}^i\right\|$ is smaller than the sensor resolution $\boldsymbol{e}_{\boldsymbol{m}}$ where $\|.\|$ is an L_2 norm, $\boldsymbol{m}_{k|k-1}^i$ would be almost unchanged. Therefore, we can terminate the particle flow step, if $\left\|\triangle\boldsymbol{m}_{k|k-1}^i\right\|<\boldsymbol{e}_{\boldsymbol{m}}$. Then the particle weight is updated by Eq. (17), where $\boldsymbol{\nabla}\boldsymbol{f}_{k,\lambda}^i$ is calculated as,

$$\boldsymbol{\nabla}\boldsymbol{f}_{k,\lambda}^i=\lambda\boldsymbol{A}\boldsymbol{\nabla}(\boldsymbol{\nabla}(\boldsymbol{B}_k^i))\boldsymbol{A}\boldsymbol{B}_k^i-\boldsymbol{A}\boldsymbol{\nabla}(\boldsymbol{B}_k^i), \tag{47}$$

where

$$\boldsymbol{A}=(\lambda\boldsymbol{\nabla}(\boldsymbol{B}_k^i)-(\boldsymbol{P}_{k|k-1}^i)^{-1})^{-1}. \tag{48}$$

$$\begin{aligned}&\boldsymbol{\nabla}(\boldsymbol{\nabla}(\boldsymbol{B}_k^i))\\&=\frac{p_{D,k}^i\boldsymbol{\nabla}^3C_k^i}{1-p_{D,k}^i+p_{D,k}^iC_k^i}-\frac{(p_{D,k}^i)^2(2\boldsymbol{\nabla}^2C_k^i\otimes\boldsymbol{\nabla}C_k^i+\boldsymbol{\nabla}C_k^i\otimes\boldsymbol{I})}{(1-p_{D,k}^i+p_{D,k}^iC_k^i)^2}\\&+\frac{2(p_{D,k}^i)^3\boldsymbol{\nabla}C_k^i\otimes\boldsymbol{\nabla}C_k^i\otimes\boldsymbol{\nabla}C_k^i}{(1-p_{D,k}^i+p_{D,k}^iC_k^i)^3},\end{aligned} \tag{49}$$

where $\otimes$ is outer product.

$$\nabla^3 C_k^i = \sum_{r=1}^{R_k} \frac{(\nabla^3 h_k^{i,r})}{\kappa_k + \sum_{i=1}^{N_k} p_{D,k}^i h_k^{i,r} \Psi_{k|k-1}^i}. \quad (50)$$

Although the differential equations such as Eqs. (47)-(50) look complex, some elements in these equations, such as $1 - p_{D,k}^i + p_{D,k}^i C_k^i$, can be stored and repeatedly used. For further reducing the computational cost, $\nabla \boldsymbol{f}_{k,\lambda}^i$ can be calculated approximately as,

$$\nabla \boldsymbol{f}_{k,\lambda}^i = \begin{cases} \frac{(\boldsymbol{f}_{k,\lambda}^i - \boldsymbol{f}_{k,\lambda-\triangle\lambda}^i)\boldsymbol{f}_{k,\lambda}^{i\,T}}{\triangle\lambda \|\boldsymbol{f}_{k,\lambda}^i\|^2} & \lambda \neq 0 \\ 0 & \lambda = 0. \end{cases} \quad (51)$$

The pseudo code of IPF in the IPF-SMC-PHD filter is shown in Algorithm 2. If the measurement and state spaces are discrete, such as in visual tracking, the differential equations such as $\nabla \boldsymbol{f}_{k,\lambda}^i$ and $\nabla(\boldsymbol{B}_k^i)$ can be calculated in terms of the gradient.

Algorithm 2 Intensity particle flow step in the IPF-SMC-PHD filter

Input: $\{\boldsymbol{m}_{k|k-1}^i, \omega_{k|k-1}^i, \boldsymbol{P}_{k|k-1}^i\}_{i=1}^{N_{k-1}}$, $\{S_k^r\}_{r=1}^{R_k}$ and $\boldsymbol{Z}_k$.
Output: $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_{k-1}}$.
Initialize: $\boldsymbol{R}$, $p_{D,k}$, $\triangle\lambda$, κ_k, v_k^i, $\partial_{\boldsymbol{m}}$ and $\boldsymbol{w}_k^i$.
for $\lambda \in [0, \triangle\lambda, 2\triangle\lambda, \cdots, N_\lambda\triangle\lambda]$ **do**
  for Each surviving particle **do**
    Calculate the likelihood density $\boldsymbol{h}_k^{i,r}$.
    Calculate $\nabla h_k^{i,r}$,$\nabla(\nabla h_k^{i,r})$ by Eq. (44) and Eq. (45).
    Calculate particle flow by Eq. (42).
    Calculate $\triangle \boldsymbol{m}_{k|k-1}^i$ by Eq. (9).
    if $\left\| \triangle \boldsymbol{m}_{k|k-1}^i \right\| < \boldsymbol{e_m}$ **then**
      Break out of the loop.
    end if
    Update each particle state by Eq. (14).
    Update the weights of the particles $\{\omega_{k|k-1}^i\}_{i=1}^{N_{k-1}}$ to obtain $\{\omega_k^i\}_{i=1}^{N_{k-1}}$ by Eq. (17).
  end for
end for
Set $\{\boldsymbol{m}_k^i\}_{i=1}^{N_{k-1}}$ as $\{\boldsymbol{m}_{k|k-1}^i\}_{i=1}^{N_{k-1}}$.

B. Birth model

In the NPF-SMC-PHD filter [39], the birth particles are generated for each measurement [46], no matter whether a new target appears or not. When no target appears, the new born particles are associated with the survival targets, and this can be computationally inefficient. To address this issue, the particle weights are associated with the prior density, and the born particles near the surviving targets are given low weights in our birth model.

For each $\boldsymbol{z}_k^r \in \boldsymbol{Z}_k$, we generate N_B newborn particles. In total, $N_B * R_k$ particles are created and the born particle set is $\{\boldsymbol{m}_k^{i,r}, \omega_k^{i,r}\}_{i=N_{k-1}+1,r=1}^{N_{k-1}+N_B,R_k}$. For each $\boldsymbol{z}_k^r$, the state of the born particle is created in terms of the Gaussian model,

$$\boldsymbol{m}_{k|k-1}^{i,r} \sim \mathcal{N}(\boldsymbol{z}_k^r, \boldsymbol{R}), \quad (52)$$

where $\boldsymbol{R}$ is the covariance matrix of the measurements. The particle weights $\omega_{k|k-1}^{i,r}$ are calculated as

$$\omega_{k|k-1}^{i,r} = \frac{S_k^{i,r}}{N_B}, \quad (53)$$

Algorithm 3 IPF-SMC-PHD Filter

Input: $\{\boldsymbol{m}_{k-1}^i, \omega_{k-1}^i, \boldsymbol{P}_{k-1}^i\}_{i=1}^{N_{k-1}}$ and $\boldsymbol{Z}_k$.
Output: $\{\tilde{\boldsymbol{m}}_k^j, \tilde{\omega}_k^j, \boldsymbol{P}_{k-1}^i\}_{j=1}^{\tilde{N}_k}$, $\{\boldsymbol{P}_k^i\}_{i=1}^{N_k}$ and $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_{k-1}}$.
Initialize: k, q_k, p_s, ϕ_k, p_k, κ_k, $p_{D,k}$, $\triangle\lambda$, N_λ, v_k^i, $\boldsymbol{w}_k^i$, $\boldsymbol{R}$, $\partial_{\boldsymbol{m}}$ and N_B.
Run:
Propagate surviving particles $\{\boldsymbol{m}_{k|k-1}^i\}_{i=1}^{N_{k-1}}$ by Eq. (4).
Calculate the particle weight $\{\omega_{k|k-1}^i\}_{i=1}^{N_{k-1}}$ by Eq. (6).
for Each measurement $\boldsymbol{z}_k^r$ **do**
  Sample N_B born particles around measurements by Eq. (52).
  Calculate $S_k^{i,r}$ by Eq. (54).
  Calculate particle weight $\omega_{k|k-1}^{i,r}$ by Eq. (53).
end for
for Each born particles **do**
  Update the weights of the particles $\omega_k^{i,r}$ by Eq. (55).
  $\boldsymbol{m}_k^{i,r} = \boldsymbol{m}_{k|k-1}^{i,r}$
end for
Calculate $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_{k-1}}$ by the IPF as the Algorithm 2.
Combine all the particles: $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_k} = \{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_{k-1}} \cup \{\boldsymbol{m}_k^{i,r}, \omega_k^{i,r}\}_{i=N_{k-1}+1,r=1}^{N_{k-1}+N_B,R_k}$.
if There are two particles with same states, $\boldsymbol{m}_{k|k-1}^i = \boldsymbol{m}_{k|k-1}^{i'}$ where $i \in \{1, ..., N_k\}$ and $i' \in \{i+1, ..., N_k\}$ **then**
  $\omega_{k|k-1}^i = \omega_{k|k-1}^i \omega_{k|k-1}^{i'}$.
  $\omega_{k|k-1}^{i'} = 0$.
end if
Calculate the estimated number of targets $\tilde{N}_k = \sum_{i=1}^{N_k} \omega_k^i$.
Cluster particles and get $\{\tilde{\boldsymbol{m}}_k^j, \tilde{\omega}_k^j\}_{j=1}^{\tilde{N}_k}$ by the k-means method or MEAP
if ESS $< N_{k-1}/2$ **then**
  Resample $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_k}$.
end if
for Each cluster **do**
  Calculate $\boldsymbol{m}_k$ and $\boldsymbol{P}_k$ by Eq. (56) and Eq. (57).
  Set $\boldsymbol{P}_k^i = \boldsymbol{P}_k$.
end for
Slot the particles $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_k}$ based on the particle weights.
Remove the $N_k - N_{k-1}$ low weight particles from $\{\boldsymbol{m}_k^i, \omega_k^i\}_{i=1}^{N_k}$.

$$S_k^{i,r} = \gamma_k(\boldsymbol{m}_{k|k-1}^{i,r}) * \max(0, 1 - \sum_{i=1}^{N_{k-1}} h_k^{i,r} \omega_{k|k-1}^{i,r}), \quad (54)$$

where $\sum_{i=1}^{N_{k-1}} h_k^{i,r} \omega_{k|k-1}^{i,r}$ is the probability of the rth measurement created by the target state $\boldsymbol{m}_{k|k-1}^i$, and $\gamma_k(\boldsymbol{m}_{k|k-1}^{i,r})$ is the probability density function of the born particle $\boldsymbol{m}_{k|k-1}^{i,r}$.

Since the estimation of the number of targets obtained from the particle weights is unstable in the SMC-PHD filter [46], $\sum_{i=1}^{N_{k-1}} h_k^{i,r}\omega_{k|k-1}^{i,r}$ may be greater than 1, especially when two targets occlude each other. Here we assume that each measurement represents only one target. $S_k^{i,r}$, which uses $\max(0, 1-\sum_{i=1}^{N_{k-1}} h_k^{i,r}\omega_{k|k-1}^{i,r})$, represents the probability of the measurements created by the new target.

Since the born particles represent the new detected targets, we assume $p_D = 1$ for the born particles and the particle state remains unchanged in the update step. The weights of the born particles are updated as:

$$\omega_k^{i,r} = \frac{h_k^{i,r}\omega_{k|k-1}^i}{\kappa_k + \sum_{j=1}^{N_k} p_{D,k}^i h_k^{i,r}\omega_{k|k-1}^i}. \tag{55}$$

The pseudo code of the IPF-SMC-PHD filter is given in Algorithm 3. Compared to Algorithm 1, the main changes are at lines 4-12 and lines 13-18 for the birth model and IPF, respectively. Apart from that, the pruning step, lines 24-28, are used to remove the repeated particles. The particles with the same states could be accurately approximated by a single particle. Hence, in practice the weights of these particles can be merged into one. In the clustering step, the cluster is denoted as $\{\boldsymbol{m}_{k|k-1}^i, \omega_{k|k-1}^i\}_{i\in\boldsymbol{\Lambda}}$, where $\boldsymbol{\Lambda}$ is a subset of $[1, \cdots, N_k]$ containing the index of particles belonging to the same cluster. The mean state $\bar{\boldsymbol{m}}_{k|k-1}$ of this particle set is given by:

$$\bar{\boldsymbol{m}}_{k|k-1} = \frac{\sum_{i\in\boldsymbol{\Lambda}} \left(\omega_{k|k-1}^i \boldsymbol{m}_{k|k-1}^i\right)}{\sum_{i\in\boldsymbol{\Lambda}} \omega_{k|k-1}^i}. \tag{56}$$

The covariance matrix $\boldsymbol{P}_{k|k-1}$ of this particle set is given by:

$$\boldsymbol{P}_{k|k-1} = \frac{\sum_{i\in\boldsymbol{\Lambda}} \left(\omega_{k|k-1}^i \boldsymbol{e}(\boldsymbol{m}_{k|k-1}^i)\boldsymbol{e}(\boldsymbol{m}_{k|k-1}^i)^T\right)}{\sum_{i\in\boldsymbol{\Lambda}} \omega_{k|k-1}^i}, \tag{57}$$

where

$$\boldsymbol{e}(\boldsymbol{m}_{k|k-1}^i) = \boldsymbol{m}_{k|k-1}^i - \bar{\boldsymbol{m}}_{k|k-1}. \tag{58}$$

For the ith particle in the subset $\boldsymbol{\Lambda}$, $\bar{\boldsymbol{m}}_{k|k-1}^i$ and $\boldsymbol{P}_{k|k-1}^i$ are set as $\bar{\boldsymbol{m}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$, respectively.

V. Numerical evaluations

We evaluate the performance of the IPF-SMC-PHD filter using both simulated data and real data, as compared with the NPF-SMC-PHD [42], ZPF-SMC-PHD [40], [41], SMC-PHD [15], auxiliary SMC-PHD (ASMC-PHD) [21], GPF-PHD [43], GLMB [56] and δ-GLMB filters [45]. For the simulated data, we consider a large number of clusters and undetected targets based on the Gaussian likelihood which may lead to severe particle degeneracy for the NPF-SMC-PHD filter due to the presence of clutter and mis-detection. For real data, we consider two datasets for multi-speaker tracking, respectively, the LOCATA dataset, which includes audio-only recordings, and the AV16.3 dataset which includes both audio and video recordings.

A. Synthetic data

In this simulation, we construct a multi-target tracking scenario, based on the setup proposed in [57]. The tracking region is 40m $\times$ 40m. Each target follows a constant dynamic model, $\mathbf{F}_{\tilde{\boldsymbol{m}}} = \begin{bmatrix} \boldsymbol{I}_2 & \triangle k\boldsymbol{I}_2 \\ \mathbf{0}_2 & \boldsymbol{I}_2 \end{bmatrix}$ where $\boldsymbol{I}_2$ and $\mathbf{0}_2$ denote the 2×2 identity and zero matrices. $\triangle k = 1$ is the sampling period. The position of the sensor is at the origin of the coordinate system.

At each time step, a detector is used to report the measurements $\boldsymbol{Z_k}$, which can be seen as a union of clutter and target measurements. The index of measurement in $\boldsymbol{Z_k}$ is irrelevant. For the measurement parameters, $\boldsymbol{R}$ is set as $\begin{bmatrix} 25\boldsymbol{I}_2 & \mathbf{0}_2 \\ \mathbf{0}_2 & 25\boldsymbol{I}_2 \end{bmatrix}$. The noise is white Gaussian $\boldsymbol{\tau}_k \sim N(0, 10)$ and $\boldsymbol{\varsigma}_k \sim N(0, 5)$. The clutter is created and spread randomly in the tracking area.

There are 4 targets and their states are $[12, 6, 0.001, 0.001]^T$, $[32, 32, 0.001, 0.005]^T$, $[20, 13, 0.1, 0.01]^T$, and $[15, 35, 0.002, 0.002]^T$. The covariance matrix is $\frac{1}{20}\begin{bmatrix} 0.33 & 0 & 0.5 & 0 \\ 0 & 0.33 & 0 & 0.5 \\ 0.5 & 0 & 1 & 0 \\ 0 & 0.5 & 0 & 1 \end{bmatrix}$.

B. Real dataset

We use both audio-only and audio-visual data in relation to multi-speaker tracking for the evaluation of our proposed algorithms. For audio-only data, there are very few publicly available datasets, and the ground-truth target position is often not provided or lies in a range of several centimeters. Recently, two audio datasets, SMARD [58] and LOCATA [59] are published. In the SMARD dataset, the single-channel and multi-channel audio are recorded. The recordings are taken in a low-reverberant room using different microphone arrays and targets. However, this dataset only considers a single source scenario. The LOCATA datasets provide a novel, comprehensive data corpus on sound source localization and tracking. It contains real-world multichannel audio recordings, obtained by hearing aids (Dummy), microphones integrated into a robot head (Benchmark2), a plana (Dicit) and a spherical microphone array (Eigenmike) in an enclosed room (7.1m$\times$9.8m$\times$3m) with 4, 12, 15 and 32 microphones, respectively. The audio signals, including measurement and ambient noise, are recorded at 48 kHz which are synchronized with the ground-truth positional data acquired by the OptiTrack system [60]. This dataset includes six tasks. In task 1, 3 and 5, there is only one target, not ideal for evaluating multi-target tracking systems. In task 2, the targets are not moving, which is not challenging for our method. In task 6, the microphone array is moving, which is complex for the multi-target tracking. In our experiment, we only test our methods on task 4, where the targets are moving and the microphone array is static. In this task, each microphone array takes three recordings of two moving targets. The measurements, i.e. the DOA lines drawn from the position of the speaker to the centre of the microphone array, are estimated by the MUSIC algorithm [61] for each frequency bin and block size of 100 frames. The MUSIC resolution is 5° in azimuth. The azimuth of DOA

lines is shown as the radian. The step-size between consecutive blocks is 10 frames.

For the audio-visual dataset, several real-world datasets are publicly available in the literature, such as the AV16.3 [62], AVDIAR [63], AVTRACK-1 [64], AVASM [65], AMI [66], CLEAR [67], MVAD [68] and SPEVI [69]. In our experiment, we used the AV16.3 dataset to evaluate the performance of our proposed algorithms, since the calibration information is provided in this dataset for the projection of the audio information from the physical space to the image plane and the dataset contains some challenging situations, e.g. the number of targets changes and some targets are occluded. AV16.3 includes the occlusion as a challenging scenario, and consists of sequences where the targets are walking and speaking at the same time. The video and audio signals are recorded by three calibrated video cameras at 25 Hz and two circular eight-element microphone arrays at 16 kHz, respectively. The audio and video streams are synchronised before running the algorithms. The size of each image frame is 288×360. Although colored balls on the head of the targets are provided for annotation purpose, they are not used in our algorithms. All algorithms are tested with all three different camera angles of four sequences: Sequences 24, 25, 30 and 45, which correspond to the cases of two to three targets. They are the most challenging sequences in term of movements of the targets and occlusions. In addition, the number of targets is varying throughout the recording. In Sequence 24 (camera 1), two targets speak and walk back and forth. They cross the field of view twice and occlude each other. In Sequence 45 (camera 3), three targets occlude each other many times. The number of targets varies from 0 to 3.

C. Performance metric

To evaluate the tracking performance, the Optimal Sub-pattern Assignment (OSPA) distance [70] and ESS are used. The OSPA between the true multi-target and estimated state is defined as,

$$\text{OSPA}(\{\tilde{\boldsymbol{m}}_k^j\}_{j=1}^{\tilde{N}_k}, \{\tilde{\mathbf{m}}_k^{\tilde{j}}\}_{\tilde{j}=1}^{\tilde{\mathfrak{N}}_k}) = \sqrt[a]{\frac{\min\limits_{\pi\in\Pi_{\tilde{\mathfrak{N}}_k,\tilde{N}_k}} \sum\limits_{j=1}^{\tilde{N}_k} \bar{d}^{(c)}(\tilde{\boldsymbol{m}}_k^j, \tilde{\mathbf{m}}_k^{\pi(j)})^a + c^a(\tilde{\mathfrak{N}}_k - \tilde{N}_k)}{\tilde{\mathfrak{N}}_k}} \tag{59}$$

where $\{\tilde{\mathbf{m}}_k^1, ..., \tilde{\mathbf{m}}_k^{\tilde{\mathfrak{N}}_k}\}$ is the set of ground truth target states, $\{\tilde{\boldsymbol{m}}_k^1, ..., \tilde{\boldsymbol{m}}_k^{\tilde{N}_k}\}$ is the set of the estimated target states and $\Pi_{\tilde{\mathfrak{N}}_k,\tilde{N}_k}$ is the set of maps $\pi : 1, ..., \tilde{N}_k \rightarrow 1, ..., \tilde{\mathfrak{N}}_k$. Here the state cardinality estimation $\tilde{N}_k$ may not be the same as the ground truth $\tilde{\mathfrak{N}}_k$. The OSPA error given in Eq. (59) is for $\tilde{N}_k \leq \tilde{\mathfrak{N}}_k$. If $\tilde{\mathfrak{N}}_k < \tilde{N}_k$, then $\text{OSPA}(\{\tilde{\boldsymbol{m}}_k^j\}_{j=1}^{\tilde{N}_k}), \{\tilde{\mathbf{m}}_k^{\tilde{j}}\}_{\tilde{j}=1}^{\tilde{\mathfrak{N}}_k}) = \text{OSPA}(\{\tilde{\mathbf{m}}_k^{\tilde{j}}\}_{\tilde{j}=1}^{\tilde{\mathfrak{N}}_k}, \{\tilde{\boldsymbol{m}}_k^j\}_{j=1}^{\tilde{N}_k}))$. The function $\bar{d}^{(c)}(\cdot)$ is defined as $\min(c, \bar{d}(\cdot))$ where c is the cut-off value, which determines the relative weighting of the penalties for the cardinality and localization errors and a is the metric order which determines the sensitivity to outliers. A lower OSPA implies a better performance. The ESS which is applied widely to evaluate the severity of the weight degeneracy problem [57], [40], [28], is given by

$$\text{ESS} = \frac{(\sum_{i=1}^{N_k} \omega_k^i)^2}{\sum_{i=1}^{N_k} (\omega_k^i)^2} \tag{60}$$

When the ESS is small, e.g. ESS $< N_k/2$, the resampling step is performed with the uniform weights. When the ESS is high, more particles are used to estimate the posterior density with an increased accuracy.

D. Parameter settings

The proposed IPF-SMC-PHD filter is compared with the NPF-SMC-PHD [42], ZPF-SMC-PHD [40], [41], ASMC-PHD [21], GPF-PHD [43], SMC-PHD [15], GLMB [56] and δ-GLMB [45] filters. The parameters for all the compared PHD filters are the same and given as in [25]. For the GLMB filters, the parameters are given as in [56]. For all the compared filters, the new birth model, which has been discussed in Section IV.B, is used and the particle flow is only used to modify the survival particle state to reduce the computational cost. We assume the proposal distribution q_k is equal to the transport distribution ϕ_k. The birth probability p_k is 0.5. The detection probability $p_{D,k}$ and the clutter density κ_k are constant and chosen based on extensive experimental studies and are found to be adequate for following simulations. For each target, we generate 50 particles. The maximum number of particles is 200. For the ZPF, NPF and IPF step, λ takes values from the set $[0, \triangle\lambda, 2\triangle\lambda, \cdots, N_\lambda\triangle\lambda]$ and $N_\lambda\triangle\lambda = 1$, where $N_\lambda = 30$ and $\triangle\lambda = 1/30$. The covariance matrix and mean value of the prior distribution are estimated by the particle sets. Resampling is performed when the ESS is smaller than $N_k/2$. 100 independent runs were performed for each compared filter.

The simulation is implemented in Matlab. The filters are tested on a computer with Intel i7-3770 CPU with a clock frequency of 3.40 GHz and 8G RAM. The operating system was Windows 7.

E. Tracking in the presence of clutter

In the first experiment, we only consider the presence of clutter. All targets are detected with the measurements consisting of clutter. Therefore we set $p_{D,k}^i = 1$ and add 20 random clutters into the measurements. In total, there are 800 clutters for all the 40 frames. The clutter states are randomly created in the tracking area. Based on [15], we set $\kappa_k = 0.0125$. One example of the simulated trajectories is shown in Fig. 1. The crosses are the starting positions of the four targets and the solid lines are their trajectories. The target states are perturbed by Gaussian noise, where the standard deviation of the measurement noise is 0.01. The blue circles show the positions of clutter. The order of targets and clutter are randomly given to the filters.

The number of targets estimated by the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters are given in Fig. 2. As the results are obtained as the average over 100 runs, the number of estimated targets is not an integer. The number of active targets is 4. Our proposed IPF-SMC-PHD filter outperforms the NPF-SMC-PHD filter in terms of

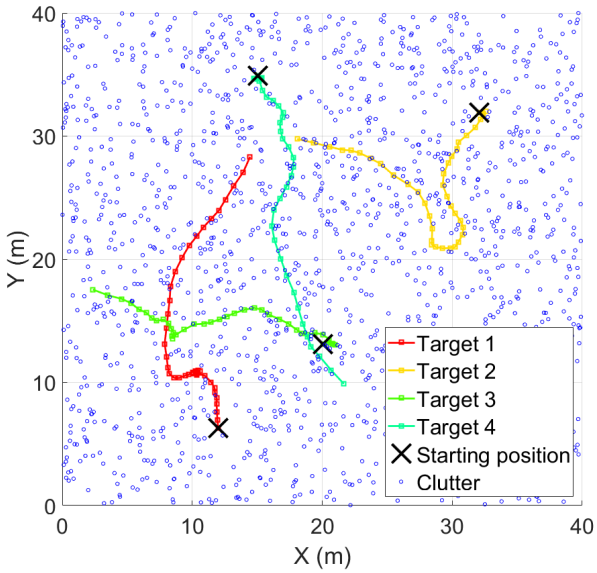

Fig. 1. The ground-truth trajectories of targets and the position of clutter.

the estimated number of targets. The IPF-SMC-PHD filter can give a correct estimate of the number of targets with the sum of the particle weights at the first frame and retains this result until the final frame.

The NPF-SMC-PHD and IPF-SMC-PHD filters give similar performance, except for frame 12 when two targets are close to each other. The NPF-SMC-PHD filter may estimate them as one target. At most of the frames, the SMC-PHD filter over-estimates the number of targets, since some clutters are estimated incorrectly as targets. In ZPF-SMC-PHD filter, the particles that are located around the nearby measurements may be considered as clutters. As a result, there may be only a few particles near the targets and the sum of particle weights could be lower than 0.5. Therefore, the ZPF-SMC-PHD filter tends to under-estimate the number of targets.

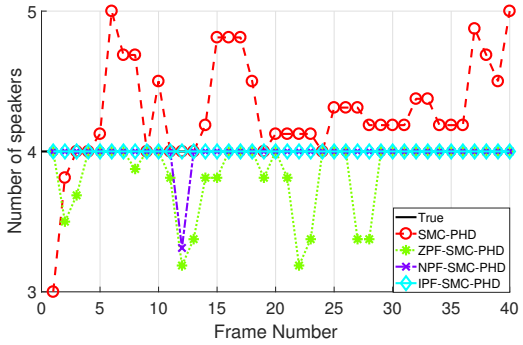

Fig. 2. The number of targets estimated by the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters when there are 20 clutter.

Fig. 3 shows the OSPA at each time step for the various tracking algorithms we compared. The OSPA of all the compared filters is increased when the number of targets is not accurately estimated. For example, the OSPA of the SMC-PHD filter is high at the first 3 frames. Among the compared methods, the IPF-SMC-PHD filter gives the lowest OSPA, i.e. the smallest average tracking error. The SMC-PHD filter has a slightly larger OSPA and average error, due to the inaccurate estimate in the number of targets. Although the IPF-SMC-PHD and NPF-SMC-PHD filters give the same estimate in the number of targets from frame 15 to frame 40, the IPF-SMC-PHD filter still gives a lower OSPA, since the particle distribution in the IPF-SMC-PHD is closer to the posterior distribution.

The proposed method is also compared with GPF-PHD, ASMC-SMC-PHD, GLMB and δ-GLMB filters, designed to mitigate the weight degeneracy problem. The average and standard deviation of OSPA and time cost are given in Table I. The IPF-SMC-PHD filter improves the estimation accuracy by 12.2%, 25.7%, 48.5%, 26.7%, 36.6% and 8.4% over the NPF-SMC-PHD, ZPF-SMC-PHD, SMC-PHD GPF-PHD, ASMC-SMC-PHD and GLMB filters, respectively. However, as the IPF considers multiple targets, the time cost of the IPF-SMC-PHD filter is increased 12% as compared with NPF-SMC-PHD filter, since the IPF considers more information, such as the detection probability and all the measurements, than the NPF. As compared to the other particle flow methods, such as the ZPF-SMC-PHD filter and NPF-SMC-PHD filter, the standard deviation of the OSPA for our proposed filters is slightly lower. Although the δ-GLMB filter has the lowest OSPA and the lowest standard deviation of OSPA, the computing time is 7.3 times higher than the IPF-SMC-PHD filter.

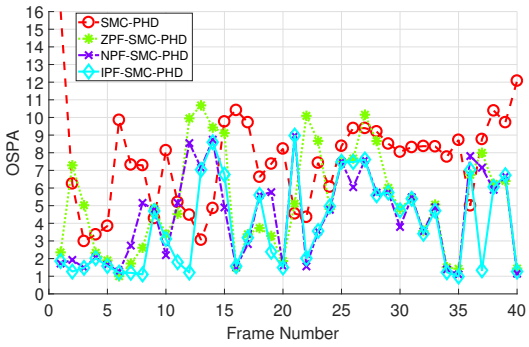

Fig. 3. The OSPA of the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters when there are 20 clutter.

TABLE I
SIMULATION TIME, AVERAGE OSPA AND STANDARD DEVIATION OF OSPA FOR THE IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD AND SMC-PHD FILTERS ON THE SIMULATED DATA.

Algorithm	Time(s)	Avg. OSPA	Std. dev.
IPF-SMC-PHD	65.7789	3.8863	1.527
ZPF-SMC-PHD	62.8198	5.2307	2.247
NPF-SMC-PHD	58.5676	4.4267	2.072
SMC-PHD	**9.5529**	7.5496	1.235
GPF-PHD	350.5448	5.3047	1.947
ASMC-PHD	36.2547	6.1287	1.6757
GLMB	153.6417	4.2447	**0.9277**
δ-GLMB	482.5714	**3.7068**	1.7208

F. Tracking in the presence of undetected targets

In this experiment, we consider the presence of disappearing targets, where the targets may not be detected with 20% probability. Therefore we set $p^i_{D,k} = 0.8$. One trial is shown in Fig. 4, where the trajectories of four targets are shown as the color-coded solid lines and the blue circles represent clutter. The positions of undetected targets are shown as the black asterisks. There are 2 random clutter in measurements and we set $\kappa_k = 0$. Other parameters are the same as in the first experiment.

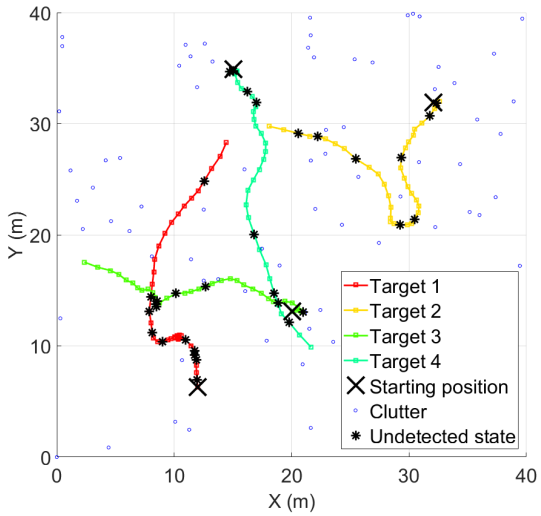

Fig. 4. The ground-truth trajectories of targets and the position of undetected targets.

The number of targets estimated by the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters is given in Fig. 5. The ground-truth number of targets is 4. Our proposed IPF-SMC-PHD filter gives a more accurate estimate of the number of targets than the NPF-SMC-PHD filter, due to the fact that the detection probability is considered. At most of the frames, the SMC-PHD filter under-estimates the number of targets, especially when some targets disappear, while ZPF-SMC-PHD over-estimates the number of estimated targets. This means when the targets disappear, the ZPF-SMC-PHD filter may treat the clutter as the targets.

Fig. 6 shows OSPA at each time step for the various tracking algorithms. For all the compared filters, when the number of targets is accurately estimated, OSPA is low. For example, the OSPA of the SMC-PHD filter is the highest among all the filters, since the estimation of the number of targets given by this method is less accurate. From the frame 5 to 10, the OSPA of the NPF-SMC-PHD filter increases, since the targets are not detected.

The average and standard deviation of OSPA and time cost of the compared filters are given in Table II. The IPF-SMC-PHD filter improves the estimation accuracy by 8%, 27%, 33%, 27%, 37% over the NPF-SMC-PHD, ZPF-SMC-PHD, SMC-PHD, GPF-PHD and ASMC-PHD filters, respectively. As the IPF considers multiple measurements, the time cost of the IPF-SMC-PHD filter is 35% higher as compared with

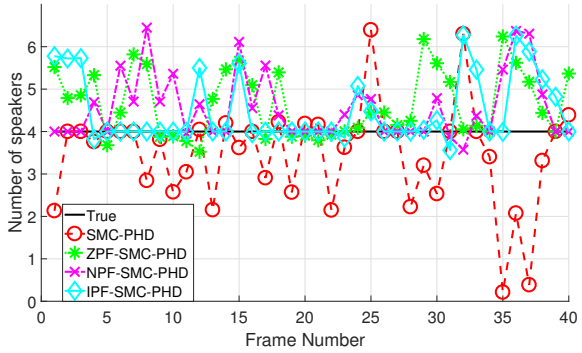

Fig. 5. The number of targets for the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters for visible and invisible targets.

that of the NPF-SMC-PHD filter. With the mapping function, GLMB filter gives the lowest standard deviation of the OSPA, while δ-GLMB gives the lowest OSPA at 7.0026 while the time cost is nearly 5 times higher than that of the IPF-SMC-PHD filter.

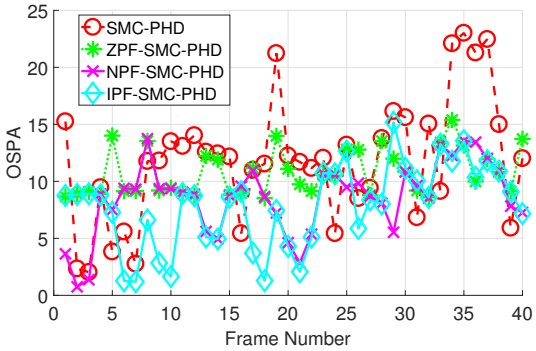

Fig. 6. The OSPA for the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters for visible and invisible targets.

TABLE II
SIMULATION TIME AND AVERAGE OSPA OF THE IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD AND SMC-PHD FILTERS ON THE SIMULATED DATA.

Algorithm	Time(s)	Avg. OSPA	Std. dev.
IPF-SMC-PHD	3.87	7.8659	2.248
ZPF-SMC-PHD	3.56	10.7617	3.581
NPF-SMC-PHD	2.86	8.5535	3.415
SMC-PHD	**0.45**	11.8482	7.2820
GPF-PHD	7.01	10.8214	2.944
ASMC-PHD	0.92	11.2254	3.254
GLMB	6.20	8.8264	**1.368**
δ-GLMB	18.64	**7.0026**	1.928

G. Tracking on the LOCATA dataset

In this subsection, we test our proposed IPF-SMC-PHD filter and baseline methods on an audio dataset, LOCATA [60]. The results in Table III show the OSPA averaged over

TABLE III
THE OSPA FOR THE IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD, SMC-PHD FILTERS AND MUSIC ALGORITHM IN TERMS OF THE OSPA ERROR ON THE LOCATA TASK 4.

Array	Recording	IPF	NPF	ZPF	SMC	MUSIC
Benchmark2	1	**1.084**	1.178	1.190	1.247	1.875
	2	**1.079**	1.165	1.187	1.242	1.753
	3	**1.093**	1.205	1.201	1.253	1.897
Dicit	1	**4.826**	5.893	5.894	7.089	10.357
	2	**4.543**	5.407	5.824	6.580	10.182
	3	**5.405**	6.777	6.874	7.860	11.057
Dummy	1	**4.833**	5.894	5.897	7.091	10.360
	2	**4.591**	5.603	5.608	6.736	9.848
	3	**5.310**	6.507	6.502	7.895	11.490
Eigenmike	1	1.465	1.559	**1.463**	1.568	2.288
	2	**1.295**	1.461	1.564	1.616	2.212
	3	**1.399**	1.503	1.542	1.656	2.429
Average OSPA		**3.077**	3.679	3.729	4.319	6.312

TABLE IV
THE OSPA OF THE AV-IPF-SMC-PHD, AV-NPF-SMC-PHD, AV-ZPF-SMC-PHD, SAVMS-SMC-PHD, AV-GPF-PHD AND AV-SMC-PHD FILTERS IN TERMS OF THE OSPA ERROR ON THE AV 16.3.

Sequence	Cam	IPF	NPF	ZPF	SMS	GPF	SMC
24	1	**10.65**	13.32	12.35	14.50	13.92	17.71
	2	**11.75**	13.20	13.24	15.35	14.58	19.83
	3	**11.64**	13.23	13.15	15.72	15.09	18.94
25	1	**12.57**	15.96	15.94	17.17	18.20	19.13
	2	**11.86**	15.29	15.21	15.39	15.14	18.47
	3	**11.52**	16.29	16.22	17.62	17.95	21.61
30	1	**12.68**	15.76	15.82	19.27	18.50	25.22
	2	**12.57**	13.41	13.43	16.16	15.19	19.37
	3	**13.25**	15.93	16.01	19.67	18.51	25.31
45	1	**16.02**	17.65	17.60	23.40	23.12	29.46
	2	**16.36**	18.60	18.55	23.16	21.71	29.47
	3	**15.96**	19.50	19.54	23.80	21.96	28.43
Average OSPA		**13.07**	15.68	15.59	18.43	17.82	22.75

each recording for task 4. In total, there are twelve recordings. For convenience, the IPF-SMC-PHD, NPF-SMC-PHD, ZPF-SMC-PHD and SMC-PHD filters are abbreviated as IPF, NPF, ZPF and SMC respectively. IPF filter has the lowest OSPA among all filters, offers performance improvements relative to the SMC, ZPF and NPF filters by 29%, 17% and 29% respectively. However, the running time of the IPF is about 6.1s per frame, nearly 2 times of the running time of the baseline SMC filter. The average OSPAs of the ZPF and NPF filter are similar at about 3.7, but the NPF (2.4s) runs faster than the ZPF (5.9s). The recording 3 for each microphone array, in which the two targets frequently occlude, is the most challenging and has the highest OSPA. Dummy and Dicit arrays have worse performance than others, since the Dummy and Dicit arrays have only two to three microphones in the vertical direction. Especially for the Dummy arrays, there are only four microphones in total and echo appears near the walls and floor. Therefore the OSPA of the Dummy array remains the highest among all the OSPAs of the different filters and approximately constant for the different recordings.

H. Tracking on AV16.3

In this subsection, we test our proposed method on the AV16.3 dataset. The proposed algorithm is compared to the baselines, which include AV-NPF-SMC-PHD [42], AV-ZPF-SMC-PHD [39], GPF-PHD [43] and sparse-AVMS-SMC-PHD (SAVMS-SMC-PHD) algorithms [71]. Note, the GPF-PHD filter is not originally proposed for target tracking [43]. In this paper, the GPF-PHD filter is modified and used as the AV-GPF-PHD filter for the AV data. Based on the Gaussian mixture model, each particle has one dependent variable for calculating zero-diffusion particle flow. The ESS for the GPF-PHD filter is calculated after pruning [43]. Other parameters are the same as the AV-SMC-PHD filter. Gebru et al. [63] also present an audio-visual Bayesian framework. As it does not apply any particle flow or PHD filter, it is not considered in our experiments.

Table IV reports the average OSPA over 10 random tests. For convenience, the AV-IPF-SMC-PHD, AV-NPF-SMC-PHD, AV-ZPF-SMC-PHD, SAVMS-SMC-PHD, AV-GPF-PHD and AV-SMC-PHD filters are abbreviated as IPF, NPF, ZPF, SMS, GPF, SMC respectively. The average error for the IPF filter is 13.07. With the contribution of the particle flow, 16% reduction in tracking error has been achieved compared with the baseline SMC filter.

We also show how significant the difference is among the results of the tested algorithms. To this end, we apply the ANOVA based F-test [72] to the results in Table IV, and obtain the significance test results, as shown in Table V. We set the significance value at 5% and the degree of freedoms in these tests at $(1, 22)$. In terms of the F-distribution table [72], the F-value, which is the ratio of the between-group variability to the within-group variability, i.e. F_{crit} for $(1, 22)$ is 4.30. The p-value is, when the null hypothesis is true, the probability of a more extreme result than the value achieved (i.e. the actual results). According to the test, the results obtained are considered statistically significant if F-value $> F_{crit}$ and p-value is smaller than the significant value (0.05). It can be seen that the improvements of the IPF, over other filters are statistically significant.

Since the measurement and state spaces are discrete for audio-visual tracking, $\nabla \boldsymbol{f}^{i}_{k,\lambda}$ could be calculated by the gradient method to reduce the computational cost. As shown in Table VI, the IPF filter only increases 12% and 58% average computational costs of the NPF and SMC filter, respectively. The computational cost of NPF and ZPF filters are similar, about 210s. The GPF filter has the highest computational cost. Although all the filters use the same initial number of particles, a few particles are duplicated in the update step of the GPF filter [43]. For all the filters, the computation cost of tracking on Sequence 45 is the highest among all sequences, since the

TABLE V
SIGNIFICANCE TEST FOR THE AV-IPF-SMC-PHD, AV-NPF-SMC-PHD, AV-ZPF-SMC-PHD, SAVMS-SMC-PHD, AV-GPF-PHD AND AV-SMC-PHD FILTERS.

method	IPF	NPF	ZPF	SMS	GPF	
IPF	-	9.70	8.55	22.46	19.78	F
	-	0.0500	0.0079	0.0010	0.0002	p-value
NPF	9.70	-	0.01	5.64	3.81	F
	0.0500	-	0.9211	0.0266	0.0638	p-value
ZPF	8.55	0.01	-	5.84	4.00	F
	0.0079	0.9211	-	0.0244	0.0580	p-value

maximum number of the targets is three in Sequence 45.

TABLE VI
COMPUTATIONAL COST IN SECONDS COMPARISON PER SEQUENCE (S/SEQUENCE) FOR THE AV-IPF-SMC-PHD, AV-NPF-SMC-PHD, AV-ZPF-SMC-PHD, SAVMS-SMC-PHD, AV-GPF-PHD AND AV-SMC-PHD FILTERS.

Sequence	IPF	NPF	ZPF	SMS	GPF	SMC
24	**129.3**	120.7	203.5	146.2	286.8	80.6
25	**130.4**	120.5	201.7	147.2	287.4	83.6
30	**131.5**	121.7	204.5	146.8	288.6	83.7
45	**197.9**	163.8	268.0	208.5	386.3	124.3
Average time	**147.2**	131.7	219.4	162.2	313.3	93.1

VI. CONCLUSION

We have presented a novel IPF-SMC-PHD filter for multi-target tracking, where particle flow with the intensity function of the clutter and the probability of detection is used to update both the states and the weights of the particles. The intensity recursion is used to replace the standard Bayes recursion of particle flow. Based on the differentiation of the intensity function with respect to the synthetic time and particle state, the intensity particle flow has been used to update the particle states and weights for multi-target tracking. Compared with ZPF and NPF, IPF can mitigate the clutters and misdetection problem by considering all the measurements in each particle flow. The IPF-SMC-PHD filter based on the Gaussian model has been tested on the simulated data, LOCATA and AV16.3 datasets. The experimental results show that the proposed filter offers a higher tracking accuracy than a number of baseline methods, including recent methods NPF-SMC-PHD, and ZPF-SMC-PHD, especially when the targets disappear or there is a larger number of clutter. Our future work will further decrease the computational cost of the IPF-SMC-PHD filter.

REFERENCES

[1] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, and S. Pankanti, "Smart video surveillance: Exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 38–51, Mar. 2005.

[2] H.-S. Yeo, B.-G. Lee, and H. Lim, "Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware," *Multimedia Tools and Applications*, vol. 74, no. 8, pp. 2687–2715, 2015.

[3] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy, "Learning to track and identify players from broadcast sports videos," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1704–1716, 2013.

[4] R. Thiolliere, E. Dunbar, G. Synnaeve, M. Versteegh, and E. Dupoux, "A hybrid dynamic time warping-deep neural network architecture for unsupervised acoustic modeling," in *Proc. INTERSPEECH*, 2015, pp. 3179–3183.

[5] P. Escudero, C. D. Bonn, R. N. Aslin, and K. E. Mulak, "Indexical and linguistic processing in infancy: Discrimination of speaker, accent and vowel differences," in *Proc. Int. Congress of Phonetic Sciences.*, May 2015, pp. 1–5.

[6] X. Qian, A. Xompero, A. Brutti, O. Lanz, M. Omologo, and A. Cavallaro, "3D mouth tracking from a compact microphone array co-located with a camera," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[7] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Courier Corporation, 2007.

[8] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition VI*, vol. 3068. International Society for Optics and Photonics, 1997, pp. 182–194.

[9] P. Del Moral, "Non-linear filtering: interacting particle resolution," *Markov processes and related fields*, vol. 2, no. 4, pp. 555–581, 1996.

[10] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.

[11] S. Blackman and R. Popoli, "Design and analysis of modern tracking systems," *Norwood, MA: Artech House.*, 1999.

[12] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[13] B.-N. Vo, S. Singh, and W. K. Ma, "Tracking multiple speakers using random sets," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2. IEEE, 2004, p. 357.

[14] R. P. Mahler, "A theoretical foundation for the stein-winter probability hypothesis density (PHD) multitarget tracking approach," DTIC Document, Tech. Rep., 2000.

[15] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo implementation of the PHD filter for multi-target tracking," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, Jul. 2003, pp. 792–799.

[16] B.-N. Vo and M. Wing-Kin, "The Gaussian mixture probability hypothesis density filter," *IEEE Trans. Signal Processing*, vol. 54, no. 11, pp. 4091–4104, Oct. 2006.

[17] R. Mahler, "A theory of PHD filters of higher order in target number," in *Proc. SPIE Defense and Security*. International Society for Optics and Photonics, 2006, pp. 62 350K–1–62 350K–12.

[18] M. A. Khan and M. Ulmke, "Non-linear and non-Gaussian state estimation using log-homotopy based particle flow filters," *Sensor Data Fusion: Trends, Solutions, Applications*, pp. 1–6, Oct. 2014.

[19] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.

[20] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. A. Wan, "The unscented particle filter," in *Advances in Neural Information Processing Systems*, 2001, pp. 584–590.

[21] N. Whiteley, S. Singh, and S. Godsill, "Auxiliary particle implementation of probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 3, pp. 1437–1454, 2010.

[22] M. R. Danaee, "Unscented auxiliary particle filter implementation of the cardinalized probability hypothesis density filter," *arXiv preprint arXiv:1506.02570*, 2015.

[23] C. Berzuini, N. G. Best, W. R. Gilks, and C. Larizza, "Dynamic conditional independence models and Markov chain Monte Carlo methods," *Journal of the American Statistical Association*, vol. 92, no. 440, pp. 1403–1412, 1997.

[24] F. Daum and J. Huang, "Particle flow and Monge-Kantorovich transport," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, 2012, pp. 135–142.

[25] V. Kilic, X. Zhong, M. Barnard, W. Wang, and J. Kittler, "Audio-visual tracking of a variable number of speakers with a random finite set approach," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, 2014, pp. 1–7.

[26] F. Daum and J. Huang, "Particle flow for nonlinear filters with log-homotopy," in *Proc. SPIE Conf. Signal Processing Sensor Fusion, Target Recognition*, vol. 6969, Apr. 2008, pp. 696 918–696 918–12.

[27] J. Heng, A. Doucet, and Y. Pokern, "Gibbs flow for approximate transport with applications to Bayesian computation," *arXiv preprint arXiv:1509.08787*, 2015.

[28] P. Bunch and S. Godsill, "Approximations of the optimal importance density using Gaussian particle flow importance sampling," *Journal of the American Statistical Association*, vol. 111, no. 514, pp. 748–762, 2016.

[29] F. Daum, J. Huang, and A. Noushin, "Exact particle flow for nonlinear filters," in *Proc. SPIE Conf. Signal Processing Sensor Fusion, Target Recognition*. International Society for Optics and Photonics, Apr. 2010, pp. 769 704–1–769 704–19.

[30] F. Daum and J. Huang, "Nonlinear filters with log-homotopy," in *Proc. IEEE Int. Conf. Information Processing of Samll Targets*. International Society for Optics and Photonics, 2007, pp. 669 918–669 918.

[31] F. Daum, J. Huang, and A. Noushin, "Coulomb's law particle flow for nonlinear filters," in *Proc. SPIE Conf. Singal and Data Processing*, O. E. Drummond, Ed. International Society for Optics and Photonics, Aug. 2011, pp. 1–15.

[32] F. Daum and J. Huang, "Zero curvature particle flow for nonlinear filters," in *Proc. SPIE Symposium on Signal and Data Processing of Small Targets*. International Society for Optics and Photonics, Apr. 2013, pp. 83 930A–83 930A–11.

[33] ——, "Particle flow with non-zero diffusion for nonlinear filters," in *Proc. SPIE Conf. Signal Processing, Sensor Fusion, and Target Recognition*, 7697, Ed., vol. 04, 2013, pp. 87 450P–87 450P–13.

[34] F. Daum, J. Huang, and A. Noushin, "Generalized Gromov method for stochastic particle flow filters," in *Proc. SPIE Conf. Signal Processing Sensor Fusion, Target Recognition*, vol. 10200. International Society for Optics and Photonics, 2017, p. 102000I.

[35] T. Ding and M. J. Coates, "Implementation of the Daum-Huang exact-flow particle filter," in *Proc. IEEE Statistical Signal Processing Workshop (SSP)*, 2012, pp. 257–260.

[36] M. A. Khan, M. Ulmke, B. Demissie, F. Govaers, and W. Koch, "Combining log-homotopy flow with tensor decomposition based solution for Fokker-Planck equation," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, 2016, pp. 2229–2236.

[37] C. Chlebek, J. Steinbring, and U. D. Hanebeck, "Progressive Gaussian filter using importance sampling and particle flow," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, 2016, pp. 2043–2049.

[38] Y. Li and M. Coates, "Particle filtering with invertible particle flow," *IEEE Trans. Signal Processing*, vol. 65, no. 15, pp. 4102–4116, 2016.

[39] Y. Liu, W. Wang, J. Chambers, V. Kilic, and A. Hilton, "Particle flow SMC-PHD filter for audio-visual multi-speaker tracking," in *Proc. IEEE Intl Conf. Latent Variable Analysis and Signal Separation*, Mar. 2017, pp. 344–353.

[40] Y. Liu, W. Wang, and Y. Zhao, "Particle flow for sequential Monte Carlo implementation of probability hypothesis density," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2017, pp. 1371–1375.

[41] Y. Liu, V. Kilic, J. Guan, and W. Wang, "Audio-visual particle flow smc-phd filtering for multi-speaker tracking," *IEEE Trans. Multimedia*, vol. 22, no. 4, pp. 934–948, Apr. 2020.

[42] Y. Liu, A. Hilton, J. Chambers, Y. Zhao, and W. Wang, "Non-zero diffusion particle flow SMC-PHD filter for audio-visual multi-speaker tracking," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[43] L. Zhao, J. Wang, Y. Li, and M. J. Coates, "Gaussian particle flow implementation of PHD filter," in *Proc. SPIE Defense and Security*, vol. 9842, May 2016, pp. 98 420D – 98 420D–10.

[44] M. Wang, H. Ji, X. Hu, and Y. Zhang, "Gaussian mixture particle flow probability hypothesis density filter," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, 2017, pp. 1–8.

[45] A.-A. Saucan, Y. Li, and M. Coates, "Particle flow superpositional GLMB filter," in *Proc. SPIE Conf. Signal Processing, Sensor Fusion, and Target Recognition*, vol. 10200. International Society for Optics and Photonics, 2017, p. 102000F.

[46] B. Ristic, D. Clark, and B.-N. Vo, "Improved SMC implementation of the PHD filter," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, Jul. 2010, pp. 1–8.

[47] T. Li, S. Sun, M. Bolić, and J. M. Corchado, "Algorithm design for parallel implementation of the SMC-PHD filter," *Signal Processing*, vol. 119, pp. 115–127, 2016.

[48] Y. Li, S. Pal, and M. J. Coates, "Invertible particle-flow-based sequential mcmc with extension to gaussian mixture noise models," *IEEE Transactions on Signal Processing*, vol. 67, no. 9, pp. 2499–2512, 2019.

[49] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. the Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[50] T. Li, S. Sun, J. M. Corchado, and M. F. Siyau, "A particle dyeing approach for track continuity for the SMC-PHD filter," in *Proc. IEEE Int. Conf. Information Fusion (FUSION)*, Jul. 2014, pp. 7–10.

[51] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputations and Bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, 1994.

[52] L. P. Kadanoff, *Statistical Physics: Statics, Dynamics and Renormalization*. World Scientific Publishing Co Inc, 2000.

[53] C. H. Edwards, *Advanced Calculus of Several Variables*. Courier Corporation, 2012.

[54] A. Vaccarella, E. De Momi, A. Enquobahrie, and G. Ferrigno, "Unscented Kalman filter based sensor fusion for robust optical and electromagnetic tracking in surgical navigation," *IEEE Trans. Instrumentation and Measurement*, vol. 62, no. 7, pp. 2067–2081, 2013.

[55] S. S. Haykin *et al.*, *Kalman Filtering and Neural Networks*. Wiley Online Library, 2001.

[56] D. Y. Kim, B.-T. Vo, and S. Nordholm, "Multiple speaker tracking with the GLMB filter," in *Control, Automation and Information Sciences (ICCAIS), 2017 International Conference on*. IEEE, 2017, pp. 38–43.

[57] Y. Li, L. Zhao, and M. Coates, "Particle flow for particle filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, May 2016, pp. 3979–3983.

[58] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third chime-speech separation and recognition challenge: Dataset, task and baselines," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 504–511.

[59] H. W. Löllmann, C. Evers, A. Schmidt, H. Mellmann, H. Barfuss, P. A. Naylor, and W. Kellermann, "The locata challenge data corpus for acoustic source localization and tracking," in *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. IEEE, 2018, pp. 410–414.

[60] ——, "The locata challenge data corpus for acoustic source localization and tracking."

[61] J. P. Dmochowski, J. Benesty, and S. Affes, "Broadband MUSIC: opportunities and challenges for multiple source localization," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2007, pp. 18–21.

[62] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez, "AV 16.3: an audio-visual corpus for speaker localization and tracking," in *Proc. Int. Workshop on Machine Learning for Multimodal Interaction*. Springer, 2004, pp. 182–195.

[63] I. D. Gebru, S. Ba, X. Li, and R. Horaud, "Audio-visual speaker diarization based on spatiotemporal Bayesian fusion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 39, pp. 7106–7113, 2017.

[64] I. D. Gebru, S. Ba, G. Evangelidis, and R. Horaud, "Tracking the active speaker based on a joint audio-visual observation model," in *Proc. IEEE Int. Conf. Computer Vision Workshops*, Dec. 2015, pp. 15–21.

[65] A. Deleforge, R. Horaud, Y. Y. Schechner, and L. Girin, "Co-localization of audio sources in images using binaural features and locally-linear regression," *IEEE/ACM Trans. Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 4, pp. 718–731, 2015.

[66] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The AMI meeting corpus: A pre-announcement," in *Proc. Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 28–39.

[67] M. H. Ooi, T. Solomon, Y. Podin, A. Mohan, W. Akin, M. A. Yusuf, S. del Sel, K. M. Kontol, B. F. Lai, D. Clear *et al.*, "Evaluation of different clinical sample types in diagnosis of human enterovirus 71-associated hand-foot-and-mouth disease," *J. Clinical Microbiology*, vol. 45, no. 6, pp. 1858–1866, Apr. 2007.

[68] V. P. Minotto, C. R. Jung, and B. Lee, "Multimodal multi-channel on-line speaker diarization using sensor fusion through SVM," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1694–1705, 2015.

[69] M. Taj, "Surveillance performance evaluation initiative (SPEVI) audio-visual people dataset," *Internet: http://www. eecs. qmul. ac. uk/ andrea/spevi. html*, 2007.

[70] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Trans. Signal Processing*, vol. 59, no. 7, pp. 3452–3457, 2011.

[71] V. Kilic, M. Barnard, W. Wang, A. Hilton, and J. Kittler, "Mean-shift and sparse sampling based SMC-PHD filtering for audio informed visual speaker tracking," *IEEE Trans. Multimedia*, vol. 18, no. 12, pp. 2417–2431, 2016.

[72] P. G. Hoel *et al.*, "Elementary Statistics," *Elementary Statistics*, 1960.