

## Research Article

# A Quantitative Analysis on Two RFS-Based Filtering Methods for Multicell Tracking

Yayun Ren<sup>1,2</sup> and Benlian Xu<sup>1</sup>

<sup>1</sup> School of Electrical & Automatic Engineering, Changshu Institute of Technology, Changshu 215500, China

<sup>2</sup> School of Information & Electrical Engineering, China University of Mining and Technology, Xuzhou 221116, China

Correspondence should be addressed to Benlian Xu; [xu\\_benlian@cslg.cn](mailto:xu_benlian@cslg.cn)

Received 15 July 2013; Accepted 6 December 2013; Published 22 January 2014

Academic Editor: Jian Li

Copyright © 2014 Y. Ren and B. Xu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multitarget filters developed from the theory of random finite sets (RFS) have recently become well-known methods for solving multitarget tracking problem. In this paper, we present two RFS-based filtering methods, Gaussian mixture probability hypothesis density (GM-PHD) filter and multi-Bernoulli filter, to quantitatively analyze their performance on tracking multiple cells in a series of low-contrast image sequences. The GM-PHD filter, under linear Gaussian assumptions on the cell dynamics and birth process, applies the PHD recursion to propagate the posterior intensity in an analytic form, while the multi-Bernoulli filter estimates the multitarget posterior density through propagating the parameters of a multi-Bernoulli RFS that approximates the posterior density of multitarget RFS. Numerous performance comparisons between the two RFS-based methods are carried out on two real cell images sequences and demonstrate that both yield satisfactory results that are in good agreement with manual tracking method.

## 1. Introduction

In previous related research in the field of medicine and biology, people often use manual visual inspection to observe living cells behaviors, such as the moving velocity and the density of cell population. Manual analysis methods always are limited by many challenges including large volume of image data and tedious manual operations. Therefore, it is of great significance to find an automatic and reliable way to track multiple cells. Over the last decades, many automatic tracking methods have been developed with rapid development of processing and computer vision technologies. Furthermore, numerous applications have been found for analyzing time-lapse cell microscopy imagery including fluorescent image [1, 2] and nonstaining cell image [3].

In the field of cell tracking, automatic and accurate tracking needs to overcome problems that mainly stem from two factors: cell factors such as population size, mitosis, and shape variability and imaging factors such as level of noise, image resolution ratio, and image contrast. The main objective of cell motion analysis is to determine the number of cells and the state of each cell, and visual multicell tracking

methods can be divided into two categories, deterministic methods and stochastic methods. Deterministic methods usually handle the detections and tracking tasks separately [4–7], but tracking may fail under problematic imaging conditions such as large cell density, cell division events, or segmentation errors.

Stochastic methods mostly fall in the category of Bayesian framework, and tracking based on Bayesian probabilistic framework methods has also been proposed in recent years and is more robust to low resolution and signal-to-noise (SNR) scenarios than other tracking methods. Successful instances include multi-cell tracking in low contrast image sequences where the number of cells varies over time, cell shape deformation, and partial overlap [8, 9].

Recently, Bayesian filtering based on Random Finite Set (RFS) theory has been proposed to solve multitarget tracking problems [10]. In the RFS formulation, the sets of targets and observations are modeled as random finite sets. This allows the problem of dynamically estimating multiple targets in presence of clutter and association uncertainty to be cast as a single-object filtering problem in a Bayesian filtering framework, where the object is the random set [10].

It is well known that this type of filter gives the optimal solution to multitarget tracking problems but is computationally intractable due to set integral calculation involved. The probabilistic hypothesis density (PHD) filter proposed by Mahler [10] overcomes this obstacle and significantly reduces the computational complexity. It is noted that the PHD recursion still involves integrals and does not have a closed-form solution. The implementation of the recursion is usually based on approximations for single-object distributions which resort to either sequential Monte Carlo method yielding sequential Monte-Carlo PHD (SMC-PHD) filter [11, 12] or the Gaussian mixture PHD (GM-PHD) filter [13, 14]. There are some reports on multitarget tracking in visual data using PHD filter. Examples include tracking of pedestrians in image sequences [15], face tracking [16], and vehicle tracking [17]. In these works, the SMC-PHD employs additional heuristics for partitioning particles to extract final state estimates and for target initialization.

The GM-PHD filter is first applied to track multiple cells in [13] which focuses on the study of cells genealogy and find that the filter could track the lineage and motion of cells relatively well. The GM-PHD filter also exhibits good performance in maintaining continuous tracks for the cells in motion, even in low contrast image [18]. A new closed-form recursion for the LGJMS-PHD filter has been presented in [19] by incorporating state-dependent transition probability and the spawned transition probability. The filter has noticeably lower processing time in the cases of numerous cells and noisy detections. Recently, a sequential Monte Carlo (SMC) implementation of multi-Bernoulli filter is presented in [8], called “track-before-detect” technique. This method by passes the detection module and makes use of the spatiotemporal information directly extracted from poor image sequences. It demonstrates very good performance in cell tracking application. Stochastic methods generally make better use of the spatiotemporal information than deterministic methods and yield more robust tracking outcomes especially for poor cell image data.

These cell tracking methods introduced above have their own properties and show different performance during implementation. This paper first quantitatively analyzes and compares the performance of GM-PHD and multi-Bernoulli filters [8, 9] in tracking multiple cells in a series of low-contrast image sequences. The cell states including cell number, position, velocity, and acceleration are estimated by both tracking methods, and the differences of obtained results are quantitatively analyzed by performance measures such as RMSE and OSPA distance. Furthermore, the detection and association errors and processing time are further discussed to compare the overall performance of the two tracking methods.

This paper is organized as follows. In Section 2, we briefly review the random finite set model in multitarget tracking problem. In Section 3, we present the principles and adapt the formulation of GM-PHD filter and multi-Bernoulli filter for multi-cell tracking. In Section 4, experimental results on real cell image sequences are presented to quantitatively analyze and compare the performance of the two methods. Finally, conclusions are summarized in Section 5.

## 2. Background on RFS Model

In a single target scenario, the state and measurement at any time are two vectors of possibly different dimensions, and these vectors evolve in time, with a constant dimension of each vector. However, in a multitarget scenario, some targets may disappear, new targets may appear, and the surviving targets may evolve to new states according to their individual dynamics. Obviously, the number and the state of targets and measurements may vary with time. In other words, the dimensions of the multitarget state and multitarget measurement evolve in time, and there is no ordering for the elements of multitarget state and measurement. Thus, such uncertainty in a multitarget system is characterized by modeling the multitarget state and multitarget measurement as random finite sets (RFS)  $\Xi_k$  and  $\Sigma_k$  on the (single target) state and observation spaces  $E_s$  and  $E_O$ , respectively. The RFS  $\Xi_k$  encapsulates all aspects of multitarget motion such as the time-varying number of targets, individual target motion, target birth, spawning, and target interactions. Given a realization  $\mathbf{X}_{k-1}$  of the RFS  $\Xi_{k-1}$  at time  $k-1$ , the multitarget state at time  $k$  is modeled by the RFS

$$\Xi_k = S_k(\mathbf{X}_{k-1}) \cup B_k(\mathbf{X}_{k-1}) \cup \Gamma_k, \quad (1)$$

where  $S_k(\mathbf{X}_{k-1})$  denotes the RFS of targets that have survived from  $\mathbf{X}_{k-1}$  according to the dynamics at time  $k$ ,  $B_k(\mathbf{X}_{k-1})$  denotes the RFS of targets spawned from  $\mathbf{X}_{k-1}$  at time  $k$ , and  $\Gamma_k$  is the RFS of targets that appear spontaneously at time  $k$ . Similarly, the RFS  $\Sigma_k$  encapsulates all sensor characteristics such as measurement noise, sensor field of view, and false alarms. Given a realization  $\mathbf{X}_k$  of  $\Xi_k$  at time  $k$ , the multitarget measurement is modeled by the RFS

$$\Sigma_k = \Theta_k(\mathbf{X}_k) \cup C_k(\mathbf{X}_k), \quad (2)$$

where  $\Theta_k(\mathbf{X}_k)$  denotes the RFS of measurements generated by  $\mathbf{X}_k$  and  $C_k(\mathbf{X}_k)$  denotes the RFS of clutter or false alarms.

The problem of multitarget filtering concerns the estimation of the multitarget state  $\mathbf{X}_k$  at time  $k$  given the set  $\mathbf{Z}_{1:k} \equiv (\mathbf{Z}_1, \dots, \mathbf{Z}_k)$  of all multitarget measurements up to time  $k$ . Let  $f_{k|k-1}(\cdot | \cdot)$  be the multitarget transition density based on the dynamic model, and let  $g_k(\cdot | \cdot)$  be the multitarget likelihood. Supposing that the multitarget posterior density  $p_{k-1|k-1}(\mathbf{X}_{k-1} | \mathbf{Z}_{1:k-1})$  is known, the posterior density at time  $k$  can be calculated recursively. The optimal multitarget Bayes filter based on RFS theory can be described by the recursion, as in [10],

$$\begin{aligned} & p_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1}) \\ &= \int f_{k|k-1}(\mathbf{X}_k | \mathbf{W}) p_{k-1|k-1}(\mathbf{W} | \mathbf{Z}_{1:k-1}) \delta \mathbf{W}, \\ & p_{k|k}(\mathbf{X}_k | \mathbf{Z}_{1:k}) = \frac{p_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1}) g_k(\mathbf{Z}_k | \mathbf{X}_k)}{\int p_{k|k-1}(\mathbf{W} | \mathbf{Z}_{1:k-1}) g_k(\mathbf{Z}_k | \mathbf{W}) \delta \mathbf{W}}, \end{aligned} \quad (3)$$

where  $p_{k|k-1}(\mathbf{X}_k | \mathbf{Z}_{1:k-1})$  is the predicted value of multitarget posterior density at time  $k$  and  $\delta$  is an appropriate reference measure on the state space.

### 3. Methods

This section provides the adaptations of the GM-PHD filter and multi-Bernoulli filter for solving multi-cell tracking problem, and the corresponding implementation issues are discussed in details as well.

**3.1. The GM-PHD Filter for Multi-Cell Tracking.** The optimal multitarget Bayes filtering recursions involve multiple integrals on multitarget state space and they are computationally intractable. The probability hypothesis density (PHD) filter, proposed by Mahler [20], helps to overcome this problem. The first statistical moment of the probability density function  $p_{k|k}(\mathbf{X}_k | \mathbf{Z}_{1:k})$ , or PHD, is denoted by  $v_{k|k}(\mathbf{x})$ , and it has a useful property that  $\int_S v_{k|k}(\mathbf{x}) d\mathbf{x}$  is equal to the expected number of targets in  $S$ . Hence, the PHD is not a probability density and does not necessarily sum to 1 but may be interpreted as a density of expected targets. The PHD filter recursion can be formulated as the prediction equation and the update equation and represented by

$$\begin{aligned} v_{k|k-1}(\mathbf{x}) &= \gamma_k(\mathbf{x}) + \int \beta_{k|k-1}(\mathbf{x} | \boldsymbol{\zeta}) v_{k-1}(\boldsymbol{\zeta}) \\ &\quad + \int p_{S,k}(\boldsymbol{\zeta}) f_{k|k-1}(\mathbf{x} | \boldsymbol{\zeta}), \\ v_k(\mathbf{x}) &= [1 - p_{D,k}(\mathbf{x})] v_{k|k-1}(\mathbf{x}) \\ &\quad + \sum_{z \in Z_k} \frac{p_{D,k}(\mathbf{x}) g_k(\mathbf{z} | \mathbf{x}) v_{k|k-1}(\mathbf{x})}{\mathbf{K}_k(\mathbf{z}) + \int p_{D,k}(\boldsymbol{\xi}) g_k(\mathbf{z} | \boldsymbol{\xi}) v_{k|k-1}(\boldsymbol{\xi}) d\boldsymbol{\xi}}, \end{aligned} \quad (5)$$

where  $p_{S,k}(\cdot)$  is the probability that a target still exists at time  $k$ ,  $p_{D,k}(\cdot)$  is the probability of detection at time  $k$ , and  $\gamma_k(\cdot)$ ,  $\beta_{k|k-1}(\cdot)$ , and  $\mathbf{K}_k(\cdot)$  denote the intensities of birth, spawn and clutter, respectively. It is noted that the PHD recursion does not admit closed-form solutions in general, and numerical integration suffers from the curse of dimensionality. The Gaussian mixture probability hypothesis density (GM-PHD) filter [14] provides a closed-form solution to the PHD filter under the assumption of linear Gaussian multitarget dynamic and measurement models and approximates the PHD and other intensities with Gaussian mixture.

In this section, we will adapt the original GM-PHD filter to solve multi-cell tracking problem, and many challenges are involved. These challenges include varying cell population densities due to cells leaving or entering the field of view and complex cellular topologies such as shape deformation, close contact, partial overlap, and uneven motions of cells. To present the GM-PHD filtering method for solving multi-cell tracking problem, we consider the following assumptions first.

- (A.1) Each cell evolves and generates observations independently of one another.
- (A.2) Clutter is Poisson and independent of cell-originated measurements.

- (A.3) The survival and detection probabilities of each cell are state independent; that is,

$$\begin{aligned} p_{S,k}(\mathbf{x}) &= p_{S,k}, \\ p_{D,k}(\mathbf{x}) &= p_{D,k}. \end{aligned} \quad (6)$$

- (A.4) The predicted multi-cell state RFS governed by  $v_{k|k-1}$  is Poisson.
- (A.5) Each cell follows a linear Gaussian dynamical model and the cell-originated measurement model has a linear Gaussian form; that is,

$$\begin{aligned} f_{k|k-1}(\mathbf{x} | \boldsymbol{\zeta}) &= N(\mathbf{x}; F_{k-1}\boldsymbol{\zeta}, Q_{k-1}), \\ g_k(\mathbf{z} | \mathbf{x}) &= N(\mathbf{z}; H_k\mathbf{x}, R_k), \end{aligned} \quad (7)$$

where  $N(\cdot; m, P)$  denotes a Gaussian density with mean  $m$  and covariance  $P$ ,  $F_{k-1}$  is the state transition matrix,  $H_k$  is the measurement matrix, and  $Q_{k-1}$  and  $R_k$  are the process noise covariance and the observation noise covariance, respectively.

- (A.6) The intensities of cell birth and spawn RFS are in the form of Gaussian mixtures as follows:

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} \omega_{\gamma,k}^{(i)} N(\mathbf{x}; m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}), \quad (8)$$

$$\beta_{k|k-1}(\mathbf{x}_k | \boldsymbol{\zeta}) = \sum_{j=1}^{J_{\beta,k}} \omega_{\beta,k}^{(j)} N(\mathbf{x}_k; F_{\beta,k-1}^{(j)}\boldsymbol{\zeta} + d_{\beta,k-1}^{(j)}, Q_{\beta,k-1}^{(j)}), \quad (9)$$

where  $J_{\gamma,k}$ ,  $\omega_{\gamma,k}^{(i)}$ ,  $m_{\gamma,k}^{(i)}$ , and  $P_{\gamma,k}^{(i)}$  are given model parameters that determine the shape of the cell birth intensity.  $J_{\beta,k}$ ,  $\omega_{\beta,k}^{(j)}$ ,  $F_{\beta,k-1}^{(j)}$ ,  $d_{\beta,k-1}^{(j)}$ , and  $Q_{\beta,k-1}^{(j)}$  determine the shape of the spawning intensity of a cell with previous state  $\boldsymbol{\zeta}$ . For simplicity, we do not consider the cell mitosis, and the corresponding spawn term  $\beta_{k|k-1}(\mathbf{x}_k | \boldsymbol{\zeta})$  is omitted here.

**Prediction Step.** Suppose that assumptions (A.1)–(A.6) hold and that the posterior intensity of multiple cells at time  $k-1$  is a Gaussian mixture of the form

$$v_{k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} N(\mathbf{x}; m_{k-1}^{(i)}, P_{k-1}^{(i)}). \quad (10)$$

Then, the predicted intensity for time  $k$  is also a Gaussian mixture [14] and is given by

$$v_{k|k-1}(\mathbf{x}) = \gamma_k(\mathbf{x}) + v_{S,k|k-1}(\mathbf{x}), \quad (11)$$

where  $\gamma_k(\mathbf{x})$  is given in (8) due to the spontaneous births such as new cells entering the field of view and  $v_{S,k|k-1}(\mathbf{x})$  is

the Gaussian mixture derived from cells that survived from the previous time step,

$$\begin{aligned} v_{S,k|k-1}(\mathbf{x}) &= p_{S,k} \sum_{j=1}^{J_{k-1}} \omega_{k-1}^{(j)} N(\mathbf{x}; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}), \\ m_{S,k|k-1}^{(j)} &= F_{k-1} m_{k-1}^{(j)}, \\ P_{S,k|k-1}^{(j)} &= Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T. \end{aligned} \quad (12)$$

*Update Step.* Suppose that assumptions (A.1)–(A.6) hold and that the predicted intensity of multi-cell at time  $k$  is a Gaussian mixture of the form

$$v_{k|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{(i)} N(\mathbf{x}; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}). \quad (13)$$

Then, the posterior intensity at time  $k$  is also a Gaussian mixture, which consists of a misdetection term and a detection term and is given by

$$v_k(\mathbf{x}) = (1 - p_{D,k}) v_{k|k-1}(\mathbf{x}) + \sum_{\mathbf{z} \in Z_k} v_{D,k}(\mathbf{x}; \mathbf{z}), \quad (14)$$

where

$$\begin{aligned} v_{D,k}(\mathbf{x}; \mathbf{z}) &= \sum_{j=1}^{J_{k|k-1}} \omega_k^{(j)}(\mathbf{z}) N(\mathbf{x}; m_{k|k}^{(j)}(\mathbf{z}), P_{k|k}^{(j)}), \\ \omega_k^{(j)}(\mathbf{z}) &= \frac{p_{D,k} \omega_{k|k-1}^{(j)}(\mathbf{z}) q_k^{(j)}(\mathbf{z})}{K_k(\mathbf{z}) + p_{D,k} \sum_{i=1}^{J_{k|k-1}} \omega_{k|k-1}^{(i)}(\mathbf{z}) q_k^{(i)}(\mathbf{z})}, \\ q_k^{(j)}(\mathbf{z}) &= N(\mathbf{z}; H_k m_{k|k-1}^{(j)}, R_k + H_k P_{k|k-1}^{(j)} H_k^T), \\ m_{k|k-1}^{(j)}(\mathbf{z}) &= m_{k|k-1}^{(j)} + K_k^{(j)}(\mathbf{z} - H_k m_{k|k-1}^{(j)}), \\ P_{k|k}^{(j)} &= [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}, \\ K_k^{(j)} &= P_{k|k-1}^{(j)} H_k^T (H_k P_{k|k-1}^{(j)} H_k^T + R_k)^{-1}. \end{aligned} \quad (15)$$

Given the Gaussian mixture intensities  $v_{k|k-1}(\mathbf{x})$  and  $v_k(\mathbf{x})$ , the corresponding expected number of cells  $\widehat{N}_{k|k-1}$  and  $\widehat{N}_k$  can be obtained by summing up the appropriate weights. Specifically, the mean of the predicted number of cells  $\widehat{N}_{k|k-1}$  is obtained by adding the mean number of birth cells and the mean number of surviving cells based on (11)

$$\widehat{N}_{k|k-1} = \sum_{i=1}^{J_{y,k}} \omega_{y,k}^{(i)} + p_{S,k} \widehat{N}_{k-1}. \quad (16)$$

Similarly, the mean of the updated number of cells  $\widehat{N}_k$  is the sum of the number of measurements and the mean number of undetected cells based on (14)

$$\widehat{N}_k = (1 - p_{D,k}) \widehat{N}_{k|k-1} + \sum_{\mathbf{z} \in Z_k} \sum_{j=1}^{J_{k|k-1}} \omega_k^{(j)}(\mathbf{z}). \quad (17)$$

It can be observed that the Gaussian mixture PHD filter suffers from computation problems due to the increasing number of Gaussian components with the evolution of the process over time. It needs pruning and merging procedures to reduce the amount of calculation and improve the efficiency of tracking. For completeness, we summarize the main steps of the GM-PHD filter in Appendix A.

*Multicell State Extraction.* In the GM-PHD filter, unlike the traditional PHD filter where the clustering process is required to extract multiobject state and usually leads to unreliable results for those objects in the proximity, the state and the number of cells can be extracted according to the Gaussian mixture representation of the posterior intensity. A good approximation to the Gaussian mixture posterior intensity

$$v_k(\mathbf{x}) = \sum_{i=1}^{J_k} \omega_k^{(i)} N(\mathbf{x}; m_k^{(i)}, P_k^{(i)}) \quad (18)$$

can be obtained after pruning and merging steps. The Gaussians with weak weights can be ignored, while the means of the Gaussians with greater weights than threshold  $T_E$  ( $\omega_k^{(i)} > T_E$ ) are selected as the output cells, and we can get the position and velocity of cell centroid in our experiments.

*3.2. The Multi-Bernoulli Filter for Multi-Cell Tracking.* The multi-Bernoulli filter recursion, proposed by Mahler [10], is a direct approximation to multitarget posterior density through propagating the parameters of a multi-Bernoulli RFS. When applied to the field of cell tracking, the multi-Bernoulli filter recursion updates a finite but time-varying number of hypothesized cell tracks, each characterized by the probability of existence and the probability density of the current hypothesized cell state.

In addition to assumptions (A.1)–(A.3) in Section 3.1, we further assume that

(A.7) cell births follow a multi-Bernoulli RFS independent of cell survivals.

*Prediction Step.* Suppose that the assumptions presented above hold and that the posterior multi-cell density at time  $k-1$  is a multi-Bernoulli of the form

$$\pi_{k-1} = \left\{ \left( r_{k-1}^{(i)}, p_{k-1}^{(i)}(\cdot) \right) \right\}_{i=1}^{M_{k-1}}, \quad (19)$$

where  $r^{(i)}$  and  $p^{(i)}(\cdot)$  denote the existence probability and the probability density of the  $i$ th independent Bernoulli RFS, respectively. Then, the predicted multi-cell intensity for time  $k$  is also a multi-Bernoulli [10] and is given by

$$\pi_{k|k-1} = \left\{ \left( r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)}(\cdot) \right) \right\}_{i=1}^{M_{k-1}} \cup \left\{ \left( r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)}(\cdot) \right) \right\}_{i=1}^{M_{\Gamma,k}}, \quad (20)$$

where  $\{(r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)}(\cdot))\}_{i=1}^{M_{\Gamma,k}}$  are the multi-Bernoulli parameters of the RFS of spontaneously born cells and the parameters  $\{(r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)}(\cdot))\}_{i=1}^{M_{k-1}}$  are given by

$$\begin{aligned} r_{k|k-1}^{(i)} &= r_{k-1}^{(i)} \langle p_{k-1}^{(i)}(\cdot), p_{S,k}(\cdot) \rangle, \\ p_{k|k-1}^{(i)}(\mathbf{x}) &= \frac{\langle f_{k|k-1}(\mathbf{x} | \cdot), p_{k-1}^{(i)}(\cdot) p_{S,k}(\cdot) \rangle}{\langle p_{k-1}^{(i)}(\cdot), p_{S,k}(\cdot) \rangle}, \end{aligned} \quad (21)$$

where  $\langle f_1(\cdot), f_2(\cdot) \rangle$  denotes the standard inner product  $\int f_1(x) f_2(x) dx$  and  $f_{k|k-1}(\cdot | \zeta)$  is the transition density at time  $k$  given previous state  $\zeta$ . In essence, the multi-Bernoulli parameter set for the predicted multi-cell density  $\pi_{k|k-1}$  is formed by the union of the multi-Bernoulli parameter sets for the surviving cells (the first term in (20)) and cell births (the second term in (20)). The total number of predicted cells is  $M_{k|k-1} = M_{k-1} + M_{\Gamma,k}$ .

*Update Step.* Suppose that the assumptions mentioned above hold and that the predicted multi-cell density for  $k$  is a multi-Bernoulli of the form

$$\pi_{k|k-1} = \left\{ \left( r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)}(\cdot) \right) \right\}_{i=1}^{M_{k|k-1}}. \quad (22)$$

Then, given by Bayes rule (4), the posterior multi-cell density can be approximated by a multi-Bernoulli with the parameters

$$\begin{aligned} r_k^{(i)} &= \frac{r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}(\cdot), g_k(\cdot) \rangle}{1 - r_{k|k-1}^{(i)} + r_{k|k-1}^{(i)} \langle p_{k|k-1}^{(i)}(\cdot), g_k(\cdot) \rangle}, \\ p_k^{(i)}(\cdot) &= \frac{p_{k|k-1}^{(i)}(\cdot) g_k(\cdot)}{\langle p_{k|k-1}^{(i)}(\cdot), g_k(\cdot) \rangle}, \end{aligned} \quad (23)$$

where  $g_k(\cdot)$  is likelihood function, which will be discussed in Section 3.3.

The sequential Monte Carlo (SMC) implementation for the multi-Bernoulli filter is briefly presented in Appendix B. Analogous to the standard particle filter, degeneracy is inevitable [21]. We resample particles for each hypothesized track after the update step to reduce the effect of degeneracy. Furthermore, the number of multi-Bernoulli parameters will grow fast as time step evolves. It needs the pruning procedure to reduce the computational load. To reduce the number of particles, pruning of hypothesized tracks is performed by discarding those with existence probabilities below a threshold  $T_p$  (we used 0.01) at each time step. In order to estimate the correct number and states of cells, the hypothesized cells with substantial overlap should be merged.

*Multicell State Extraction.* According to the parameters of multi-Bernoulli RFS representing the multi-cell state after pruning and merging steps, the number of cells and their states can be estimated via finding the existence probabilities that are larger than a threshold (we used 0.5 in our experiments). In an alternative approach, the estimated number of cells is the cardinality mean or mode, and the cell individual state can be calculated by the weighted average of the particles of the corresponding density.

### 3.3. Implementation Issues

*3.3.1. Issues in GM-PHD Filter.* As implicated in (14), the measurement values from each frame are required to compute the means, covariance, and weights of  $v_k(\mathbf{x})$  using (15). To calculate the measurement values, we propose a stable and robust multi-cell detection method in our studied data images. First, we use the grey level transformation to enhance the quality of the original image in order to facilitate separating cells from the image background. After the image preprocessing, we employ the image binarization processing. Since cell regions are darker than the background in our experiments, image inversion should be operated on the binary image. In order to make the contour of connected domain closer to the real cell, mathematical morphology processing, that is, close operation, is employed to compute the area, centroid, length, and width of each connected component. Finally, the output cell measurements can be calculated by ignoring the connected components with meaningless area and length-breadth ratio. As an example, Figure 1 shows the image processing results for a given real cell image using our proposed method and it can be observed that cell measurements, that is, the centroids of three connected components in Figure 1(f), can be effectively and accurately detected.

Our experiments focus on maintaining continuous tracks for multi-cell in motion. Therefore, we assume that the cells follow linear Gaussian dynamics and measurement models as (7), respectively, and each state of cell is represented by

$$\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^T, \quad (24)$$

where  $(x_k, y_k)$  denotes the position of cell centroid and  $(\dot{x}_k, \dot{y}_k)$  denotes the corresponding cell velocity at time  $k$ . Furthermore, the motion and observation process can be modeled as

$$\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{u}_k, \quad (25)$$

$$\mathbf{z}_{k+1} = H_{k+1} \mathbf{x}_{k+1} + \mathbf{v}_{k+1},$$

where we have

$$F_k = \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (26)$$

and  $\Delta$  is the sampling period. Furthermore,  $\mathbf{u}_k \sim N(0, Q_k)$  and  $\mathbf{v}_k \sim N(0, R_k)$  are the process noise vector and the observation noise vector, respectively, and the covariance matrices  $Q_k$  and  $R_k$  of the process and measurement noise are given by

$$Q_k = \begin{bmatrix} B & 0_2 \\ 0_2 & B \end{bmatrix} \sigma_v^2, \quad R_k = \sigma_w^2 I_2, \quad (27)$$

where  $B = \begin{bmatrix} \Delta^3/3 & \Delta^2/2 \\ \Delta^2/2 & \Delta \end{bmatrix}$ ,  $0_n$  and  $I_n$  denote the  $n \times n$  zero and identity matrices, respectively, and  $\sigma_v$  and  $\sigma_w$  are the standard deviations of the process and measurement noise, respectively. The taken values of these parameters for multi-cell tracking are determined empirically according to the dynamics of cells (we set  $\sigma_v = 1$  pixel and  $\sigma_w = 1.5$  pixels in our experiments).

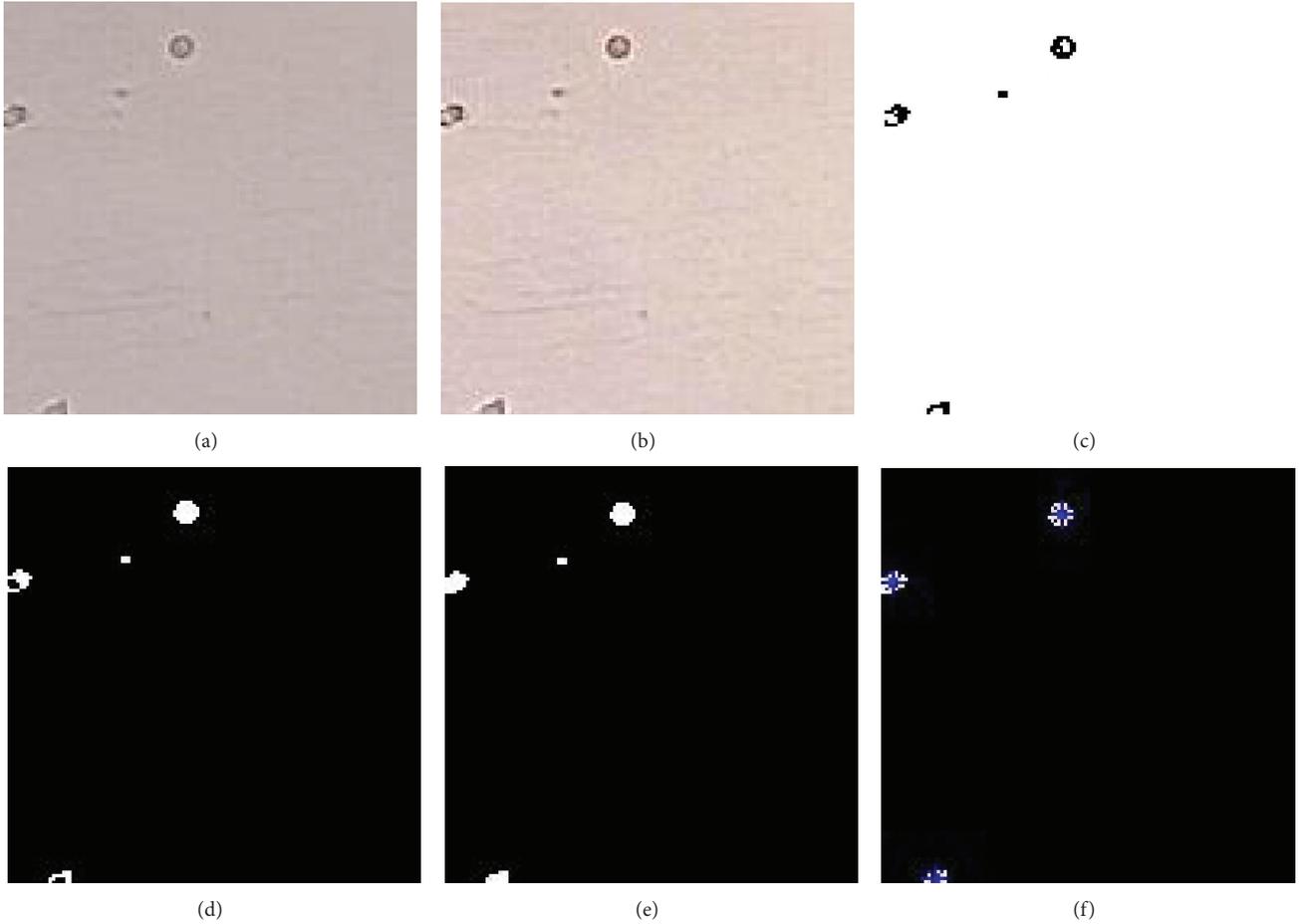


FIGURE 1: Cell detection results. (a) Original cell image. (b) Grey level transformation. (c) Image binarization. (d) Image inversion. (e) Closed operation. (f) Output measurement values.

**3.3.2. Issues in Multi-Bernoulli Filter.** The multi-Bernoulli filter, also called a “track-before-detect” method in [8], by passed the detection module and exploited the spatiotemporal information directly from the image sequence to estimate the multiobject state. The key step is to model the measurement model and to design the likelihood function  $g_k(\cdot | \cdot)$  in (4). Let  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  denote the multi-cell state vectors, and let  $\mathbf{y}_k = [y_{1,k}, y_{2,k}, \dots, y_{m,k}]$  denote the image observation comprising an array of  $m$  pixels (or bins) value  $y_{i,k}$  of the  $i$ th pixel, which can be a real number or a vector depending on the application. If each cell or state is represented by a rectangular blob, then the corresponding histogram  $v_i$  based on the cell state  $\mathbf{x}_i$  can be calculated ( $i = 1, \dots, n$ ). Based on the histogram difference comparison between the interested blob region and the cell training database, the similarity between the candidate and real cell can be estimated by (28) using total  $T_k$  cell training vectors  $\{v_j^*\}_{j=1}^{T_k}$ . We have

$$g_k(\mathbf{y}_k | \mathbf{x}_i) = \left( e^{-(1-d(v_i))^\gamma} \right)^\xi, \quad (28)$$

$$d(v_i) = \frac{1}{T_k} \sum_{j=1}^{T_k} \sum_{l=1}^C \min(v_i(l), v_j^*(l)), \quad (29)$$

where  $\gamma$  and  $\xi$  are the adjustment coefficients designed for achieving better likelihood discrimination,  $v_i(l)$  denotes the value of the  $l$ th element of the color histogram vector  $v_i$ , and  $C$  is the total number of elements in a histogram vector and takes the values of  $256 \times 3$  for an RGB image.

In addition, it is observed that the shape and brightness of cells might change over time. In order to reduce the negative effect of this interference, the cell training template should be updated in time. Specifically, assume that the cell template  $\mathbf{t}_k$  has total  $T_k$  vectors  $\{v_j^*\}_{j=1}^{T_k}$  at time step  $k$ , and if the similarity between the potential and template is greater than a threshold  $T_0$  (i.e.,  $g_k(\mathbf{t}_k | \hat{\mathbf{x}}) > T_0$ ), the potential  $\hat{\mathbf{x}}$  is added to the cell template resulting as  $\mathbf{t}_{k+1} = \{\mathbf{t}_k, \hat{\mathbf{x}}\}$ . Obviously, the dimension of the cell template gradually increases with the update process, and we set  $T_{\max} = 10$  to limit the dimension of cell template in our experiments for the sake of easy implementation and less computational burden.

**3.3.3. Label Management.** The common approach of label management for the GM-PHD filter and multi-Bernoulli filter is briefly described as follows. Each Gaussian component or Bernoulli particle at time  $k$  simply inherits the label of its predecessor (in prediction step) if it belongs to one of

the first predicted  $M_{k-1}$  Gaussian components or Bernoulli particles; otherwise, it belongs to a newly born target and is assigned a new label, and the labels remain unchanged during the update step of the filter [10, 22]. However, this method works best when the targets are well separated. In fact, cell movement is random and changeable over time. When two cells occlude each other, they are easily merged in one frame and separated in another frame. As a result, one of them would be detected by the filter as a newly born cell, and would be assigned a new label as a new track. To solve the label management problem, we present a method developed from a graph theoretical framework in which the previously missed cells are taken into account besides the most recent cells, for their association with currently detected cells [10]. This label management scheme has a memory to keep the labels of missed cells for later use in case they reappear. The recorded labels will only be removed from memory if they are missed for several consecutive frames. This method employs a three-stage constrained nearest neighbor graph search to find candidate tracks in a bipartite graph. The search is constrained in the sense that an edge is chosen only if its cost is less than a certain limit [8].

*Initialization.* The initialization takes place in the first frame. The estimates of locations of cells are read and labeled and play the role of previous cells for the next frame. After initialization, an iterative labeling contained three-stage constrained nearest neighbor graph search and memory update is repeated for each of the next frames.

*Stage 1.* Some of the current estimated cells are matched to some of the estimates from the previous iteration.

*Stage 2.* The remainder of the current estimates is matched to some previous cells that have been missed at least once.

*Stage 3.* The remainder of the current estimates is checked if they are close enough to the boundaries of the image, and if validated, they are assigned new labels.

*Memory Update.* The previous cells that have been missed for too long are removed from the memory, and other identity data is updated for the next iteration.

## 4. Experiments and Discussions

In this section we will apply the above presented two RFS-based tracking methods on two real cell image sequences and quantitatively analyze and compare their performance in different scenarios. First, we present several multicell-related measures to evaluate the tracking performance in Section 4.1. Then the analysis and comparison of the results of two experiments are presented in Section 4.2.

*4.1. Performance Measures.* (1) Root mean square error (RMSE): due to lack of ground truth for real data, we evaluate the accuracy performance of tracking by comparison with manual tracking results using a traditional quantitative

performance measure, that is, the root mean square error (RMSE):

$$\text{RMSE}_k = \sqrt{\frac{1}{\text{IR}} \sum_{i=1}^{\text{IR}} \text{RMSE}_{i,k}^2} \quad (30)$$

with

$$\text{RMSE}_{i,k}^2 = \frac{1}{M} \sum_{m=1}^M \|x_{m,k} - \hat{x}_{m,k}\|^2, \quad (31)$$

where IR is the total number of successfully tracking results (we used IR = 10 in our experiments),  $M$  is the total number of cells at step  $k$ , and  $x_{m,k}$  and  $\hat{x}_{m,k}$  are defined as the true and estimate positions of the cell  $m$  at time step  $k$ , respectively.

(2) Optimal subpattern assignment (OSPA): besides the RMSE measure, in multiobject tracking system, we usually use an alternative metric OSPA designed under the basis of Wasserstein distance (WD), proposed in [23]. It overcomes some drawbacks by introducing a parameter which controls the relative emphasis of localizations and cardinality errors, and can accommodate cardinality differences in a mathematically consistent and physically meaningful sense. The definition of OSPA metric is briefly described as follows. Let  $d^{(c)}(x, y) \triangleq \min(c, d(x, y))$  denote the distance between  $x$  and  $y$  with a nonnegative cut-off parameter  $c$  (we used  $c = 100$  in our experiments);  $\Psi_k$  is the set of permutations on  $\{1, 2, \dots, k\}$  for any  $k \in \mathbb{N} = \{1, 2, \dots\}$ . For  $1 \leq p < \infty$ ,  $m, n \geq 0$  and any two finite subsets  $\mathbf{X} = \{x_1, x_2, \dots, x_m\}$  and  $\mathbf{Y} = \{y_1, y_2, \dots, y_n\}$ , if  $m \leq n$ , the OSPA is then defined as

$$\bar{d}_p^{(c)}(\mathbf{X}, \mathbf{Y}) \triangleq \left( \frac{1}{n} \left( \min_{\psi \in \Psi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\psi(i)})^p + c^p(n-m) \right) \right)^{1/p}, \quad (32)$$

and  $\bar{d}_p^{(c)}(\mathbf{X}, \mathbf{Y}) = \bar{d}_p^{(c)}(\mathbf{Y}, \mathbf{X})$  otherwise. In addition, if  $p \rightarrow \infty$ , the OSPA metric is

$$\bar{d}_\infty^{(c)}(\mathbf{X}, \mathbf{Y}) \triangleq \begin{cases} \min_{\psi \in \Psi_n} \max_{1 \leq i \leq n} \bar{d}^{(c)}(x_i, y_{\psi(i)}) & \text{if } m = n, \\ c & \text{otherwise.} \end{cases} \quad (33)$$

Note that if  $m = n = 0$  we set the distance to 0 directly for both cases.

(3) Detection and association errors: there are some other measures to evaluate the performance of cell detection and association such as false negative rate (FNR), false alarm rate (FAR), label switching rate (LSR), and lost tracks ratio (LTR). FNR and FAR quantify the detection performance, defined as the total number of cells that are missed and the total number of nonexisting cells that are detected, respectively, both normalized over the accumulative total number of true cells over all frames. LSR and LTR quantify the association performance. LSR is the number of label switching events normalized over total number of ground truth tracks crossing events, while LTR is the number of tracks lost for more than 50% of their lifetime normalized over total number of ground truth tracks.

*4.2. Experiments on Real Cell Image Sequences.* We evaluate the performances of the two RFS-based methods on two real low-contrast multi-cell image sequences including different cell dynamics, cell morphology (shape) variation, closely moving cells, and varying number of cells in different frames. In all cases, we assume a constant survival probability  $p_S = 0.99$  and a constant detection probability  $p_D = 0.98$ . For the GM-PHD filter, we consider a predefined model for birth Gaussian components  $\gamma_k$ , that is, we divide the cell image into nine independent parts and Gaussian components are initialized with the location being uniformly distributed within each part and with a random velocity located within  $[-5 \text{ pixels}, 5 \text{ pixels}]$ , and the variance  $P_{\gamma,k} = \text{diag}(2 \text{ pixels}, 0.2 \text{ pixel/s}, 2 \text{ pixels}, 0.2 \text{ pixel/s})$  and the weight of each Gaussian component is a random number. Moreover, the number of Gaussian components of each part is a random number between 100 and 200; the standard deviation of the Gaussian observation model  $R_k = \text{diag}(2 \text{ pixels}, 2 \text{ pixels})$  and the intensity of clutter is set to  $10^{-6}$ . For the multi-Bernoulli filter, we consider a predefined model for birth Bernoulli terms denoted by known parameters  $\{(r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)})\}_{i=1}^{M_{\Gamma,k}}$  where the density  $p_{\Gamma,k}^{(i)}(\cdot)$  is represented by the particles  $\{(w_{\Gamma,k}^{(i,j)}, \mathbf{x}_{\Gamma,k}^{(i,j)})\}_{j=1}^{L_{\Gamma}}$ ; in our experiments, we assume that one cell appears in each of nine parts of the image with the location of the particle being uniformly distributed within the part (i.e.,  $M_{\Gamma,k} = 9$ ) and with a random velocity located within  $[-5 \text{ pixels}, 5 \text{ pixels}]$ , and set  $r_{\Gamma}^{(1)} = \dots = r_{\Gamma}^{(9)} = 0.1$  and  $L_{\Gamma}$  is a random number between 100 and 200; furthermore, the process noise is a Gaussian variable with zero mean and variance  $P_k = \text{diag}(2 \text{ pixels}, 0.2 \text{ pixel/s}, 2 \text{ pixels}, 0.2 \text{ pixel/s})$ ; the initial cell template  $\mathbf{t}_1$  is determined according to cells in the first frame and keeps updated using the method discussed in Section 3.3.

*Case 1.* The sequences are  $128 \times 128$  pixels RGB image with 35 frames, and the sampling period of image is  $\Delta = 4 \text{ s}$ . Each cell is indicated by the rectangle  $10 \times 10$  pixels, as shown in Figure 2, and it can be observed that the upper cell's displacement is relatively smaller and moves with nearly round shape, while the lower cell has a larger displacement and moves with irregular shape across the entire image and cell 3 partially enters the image from the lower boundary at frame 35. Figure 3 gives the position estimates of each cell in  $x$  and  $y$  directions, respectively, which are compared with the results of manual tracking method. Figure 4 plots the velocity and acceleration estimates of cell 2 using different methods. Figures 2 to 4 show that the GM-PHD filtering method and multi-Bernoulli filtering method can both automatically track each cell; moreover, the velocity and acceleration of cell can be roughly estimated. Figure 5 plots the RMSE curves ( $IR = 10$ ) and shows that the average errors of the GM-PHD filter are 0.20 pixel and 0.16 pixel in  $x$  and  $y$  directions, respectively, while 1.15 pixels and 1.31 pixels are computed for the multi-Bernoulli filtering method. Obviously, both methods have high accuracy, but the RMSE of the multi-Bernoulli filtering method is more than six times that of the former.

TABLE 1: Performance comparison results.

Methods	FAR (%)	FNR (%)	LSR (%)	LTR (%)
GM-PHD	0	2.70	0	0
Multi-Bernoulli	1.13	5.55	7.00	0.50

Figure 6 shows the estimated average number of cells in 10 independent Monte-Carlo runs against the true cell number along with the corresponding OSPA distance. It can be observed that the OSPA distance becomes larger for penalizing such discrepancy when the estimated number of cell does not coincide with the true one (such as the OSPA distance varying from 0.56 to 70.72 in frame 7 using the GM-PHD filter), and the OSPA distance remains a small value if the estimated number of cell is equal to the true one. In frames 7 and 17, the estimated number by the GM-PHD filtering method is always larger than the true numbers due to clutters. And the errors of estimated number by the multi-Bernoulli filtering method often happen at frame 35 and between frame 19 and frame 27. The average OSPA distance of each filter is 7.50 and 15.03, respectively. Through comparative analysis, it can be observed that the average OSPA distance of the GM-PHD filtering method is only half that of the multi-Bernoulli filtering method; that is, the performance using the GM-PHD filter is increased by 100.4%.

We record all false negative reports, false alarm reports, label switching reports, and tracks lost reports in each frame over 100 Monte-Carlo simulations, and the averaged FNR, FAR, LSR, and LTR are computed as presented in Table 1. We find there is only a small amount of false negative reports in frame 1 or frame 2 when using the GM-PHD filter, and the multi-Bernoulli filter is insensitive to cell 3 in Figure 2. By comparing the results, the GM-PHD filtering method performs better than the multi-Bernoulli filtering method. Finally, the total time of treatment is an important performance index for multi-cell tracking. We calculate the running time of each module such as detection module, prediction module, update module, plot module, and others (including pruning, merging, and label management). As shown in Figure 7, the average total time of treatment for the GM-PHD filter and the multi-Bernoulli filter is 38.6118 s and 30.0957 s, respectively (averaged over 10 independent runs), and it is noted that most of computation has been spent in the update module.

*Case 2.* In this case, the sequences are  $201 \times 201$  pixels RGB image with 20 frames. Each cell is indicated by the rectangle  $15 \times 15$  pixels, and the sampling period of image is  $\Delta = 3 \text{ s}$ . Figure 8 presents an example of successfully tracking results using the GM-PHD filtering method and the multi-Bernoulli filtering method, respectively. It can be observed that two methods both track at most 7 cells over the scene and one cell disappears in frame 12 and three cells (i.e., cell 5, cell 6, and cell 7) sequentially enter the scene from the boundary of frames 2, 12, and 18, respectively. Figure 9 plots the corresponding position estimate of each cell in  $x$  and  $y$  directions per frame and indicates that two methods have different sensitivity to some cells which are partially entering

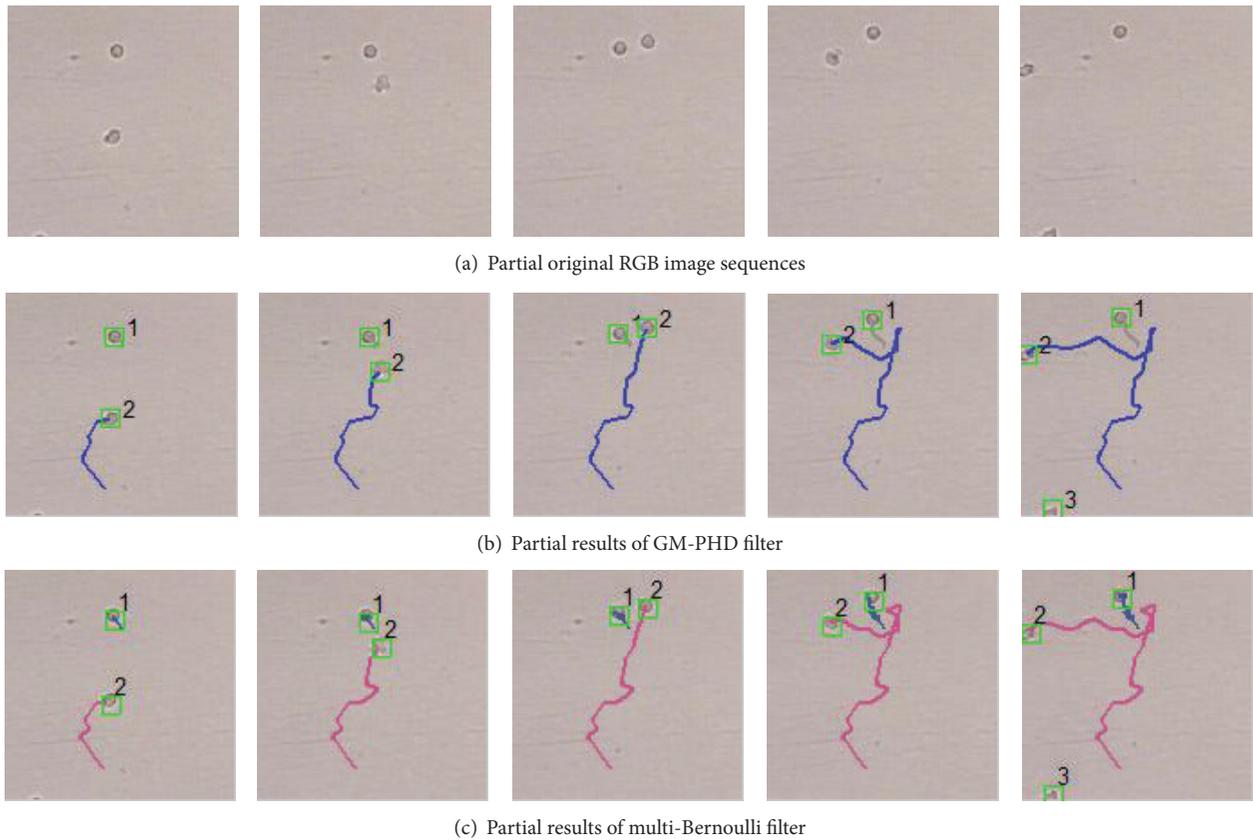


FIGURE 2: Tracking partial results of original RGB image sequences, respectively, frame 9, 19, 22, 30, and 35.

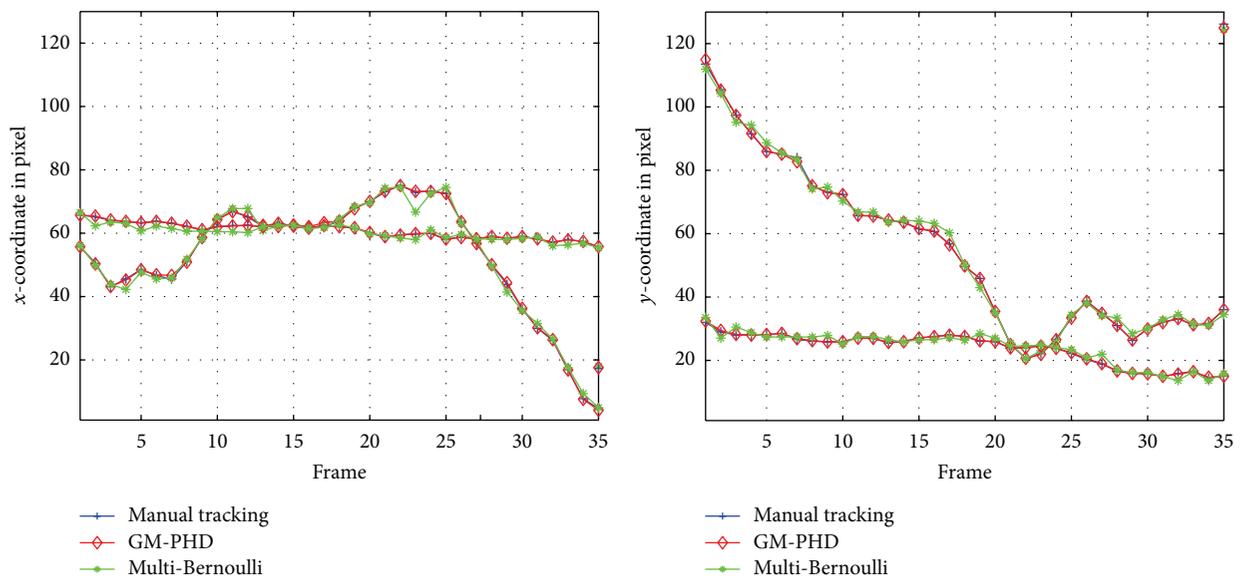


FIGURE 3: The corresponding positions of each cell in  $x$  and  $y$  directions per frame using different methods.

or leaving the image such as cell 5 and cell 6. Figure 10 plots one cell's instant velocity and acceleration estimates by different filters. It can be observed that the GM-PHD filter constantly corrects the velocity to make cell close to the true speed. For the multi-Bernoulli filter, although it can

make the speed close to the true roughly, there is a bigger fluctuation than the GM-PHD filter.

Figure 11 plots the RMSE curves in cell position estimations. Although the RMSE using the GM-PHD filter is unexpectedly larger in frames 8 and 11 due to the unsatisfactory

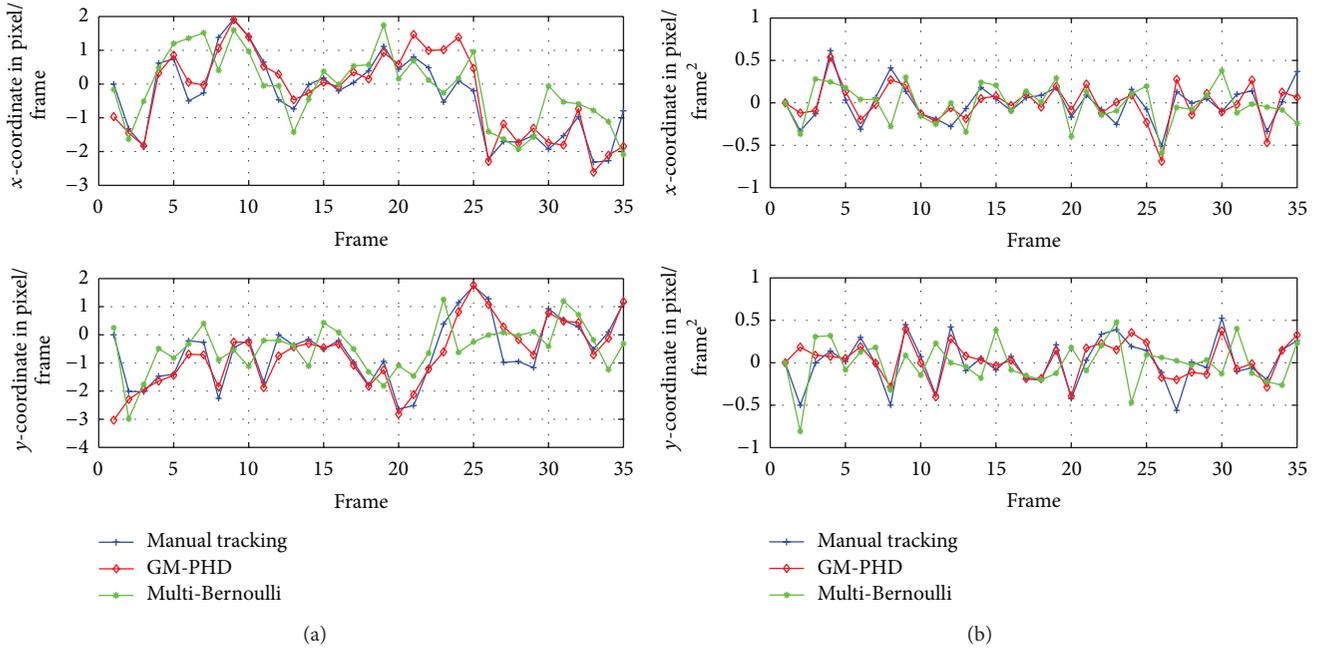


FIGURE 4: Cell instant velocity (a) and acceleration (b) estimates (cell 2 in Figure 2).

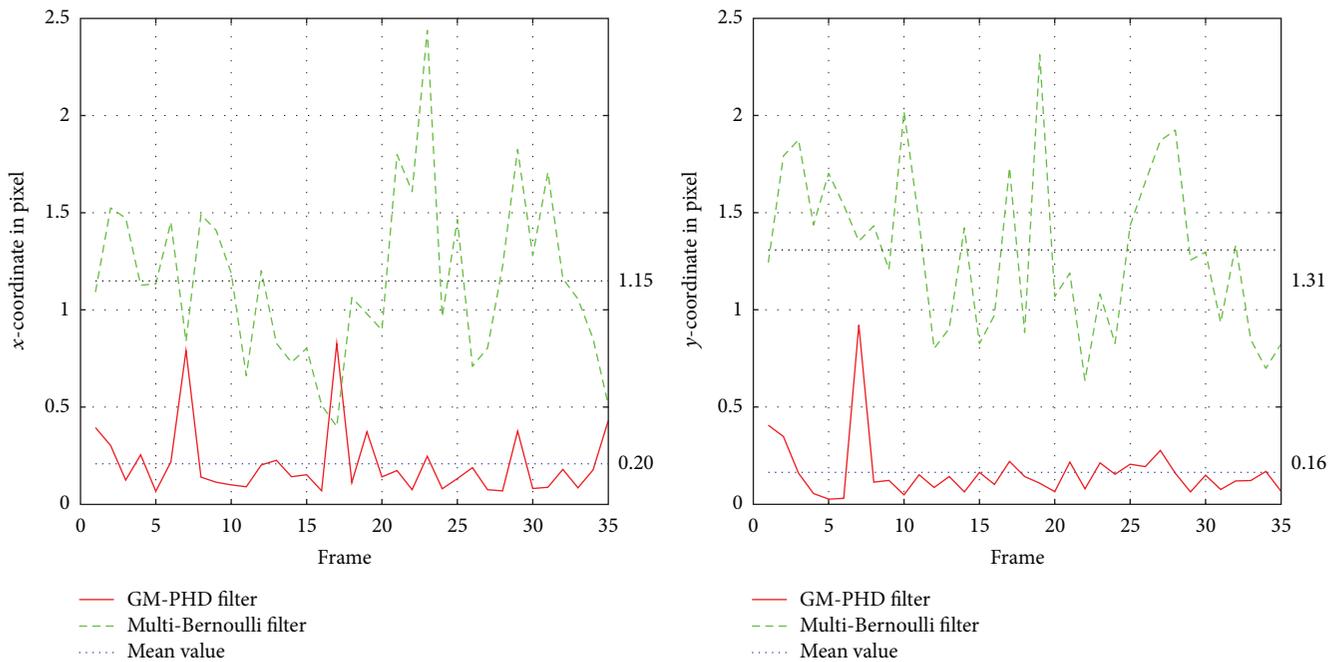


FIGURE 5: RMSE in cell position estimation using the GM-PHD filter and multi-Bernoulli filter.

measurement of detection, the accuracy of the GM-PHD filter is globally superior to the multi-Bernoulli filter, specifically improved by 343.5%. Figure 12 shows the estimated average number of cells in 10 independent runs.

Monte-Carlo runs against the true cell number along with the corresponding OSPA distance. It can be observed that in frames 10 and 11 the estimated cell numbers using two methods are not consistent with the true number due to

cell 6 looming in the image boundary and result in large OSPA distance in these frames. The average OSPA distance of each filter is 6.27 and 21.89, respectively, and we can deduce that the performance of the GM-PHD filter is more than 3 times higher than the multi-Bernoulli filter. Furthermore, the averaged FNR, FAR, LSR, and LTR once again prove the superior performance of the GM-PHD filter in Table 2. Figure 13 records the average total time of treatment, 70.9262 s and

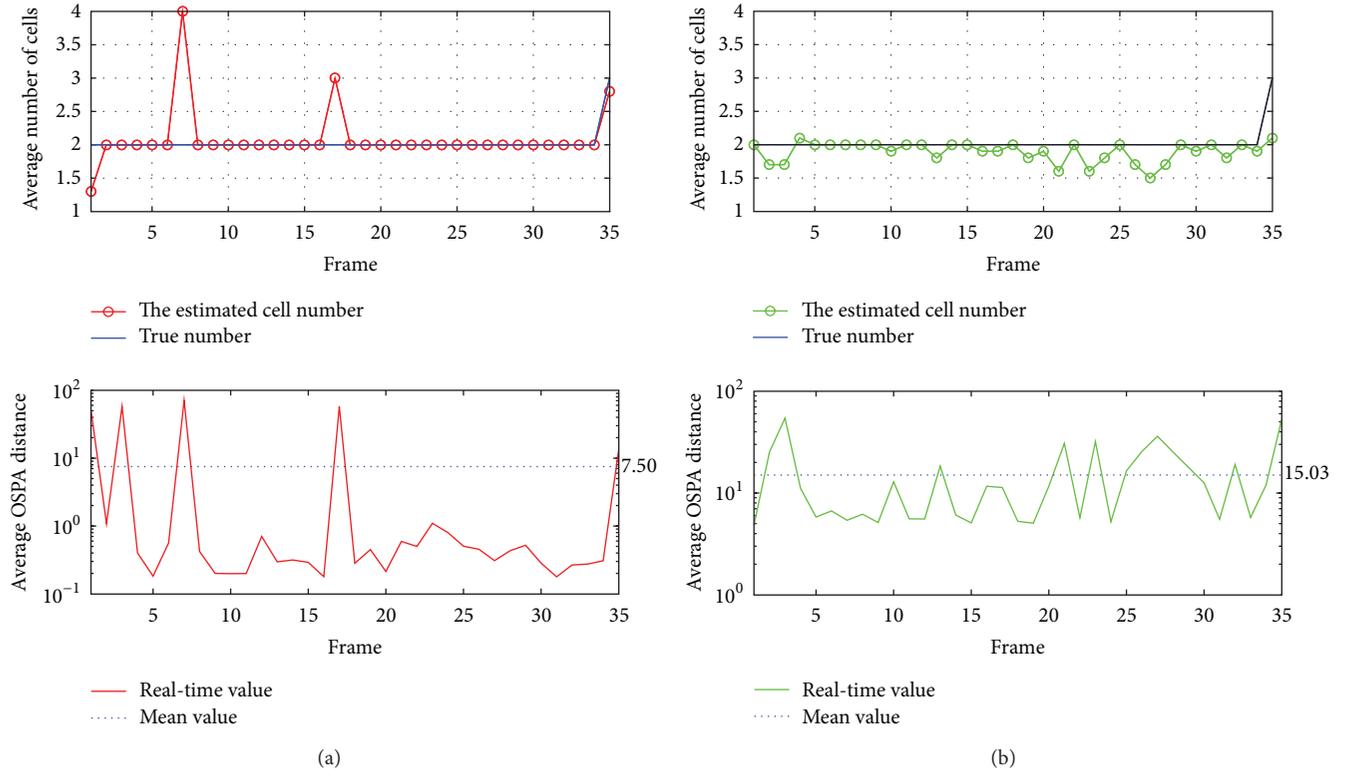


FIGURE 6: Cell number estimate and OSPA distance of two methods. ((a) GM-PHD filter and (b) multi-Bernoulli filter).

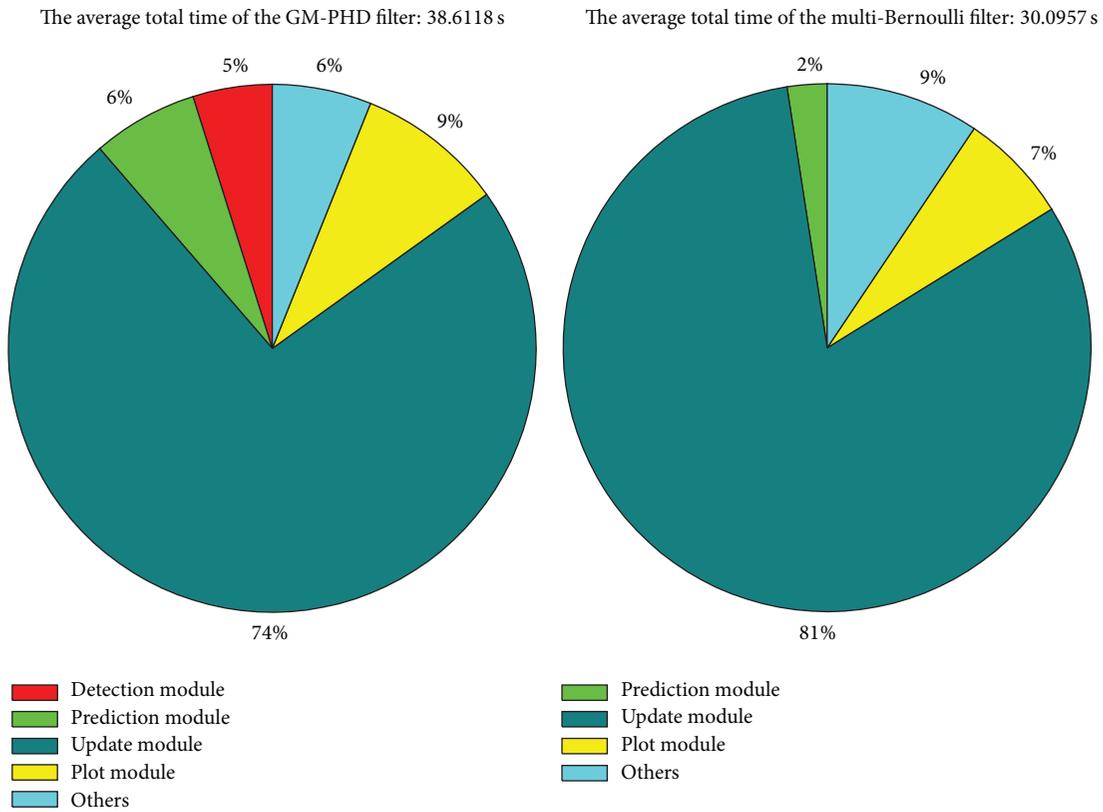


FIGURE 7: The average total time of treatment to different methods (10 independent runs).

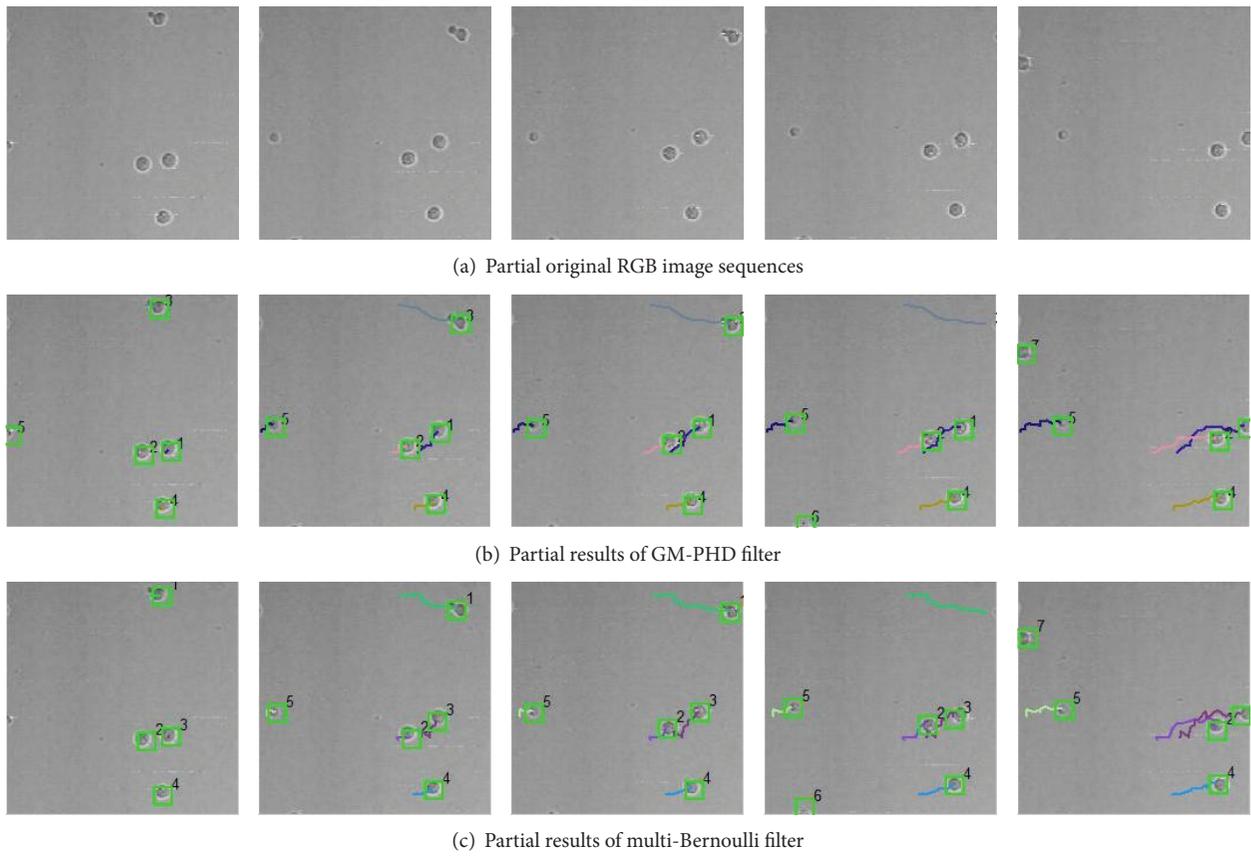


FIGURE 8: Tracking partial results of original RGB image sequences, respectively, frame 2, 8, 10, 12, and 18.

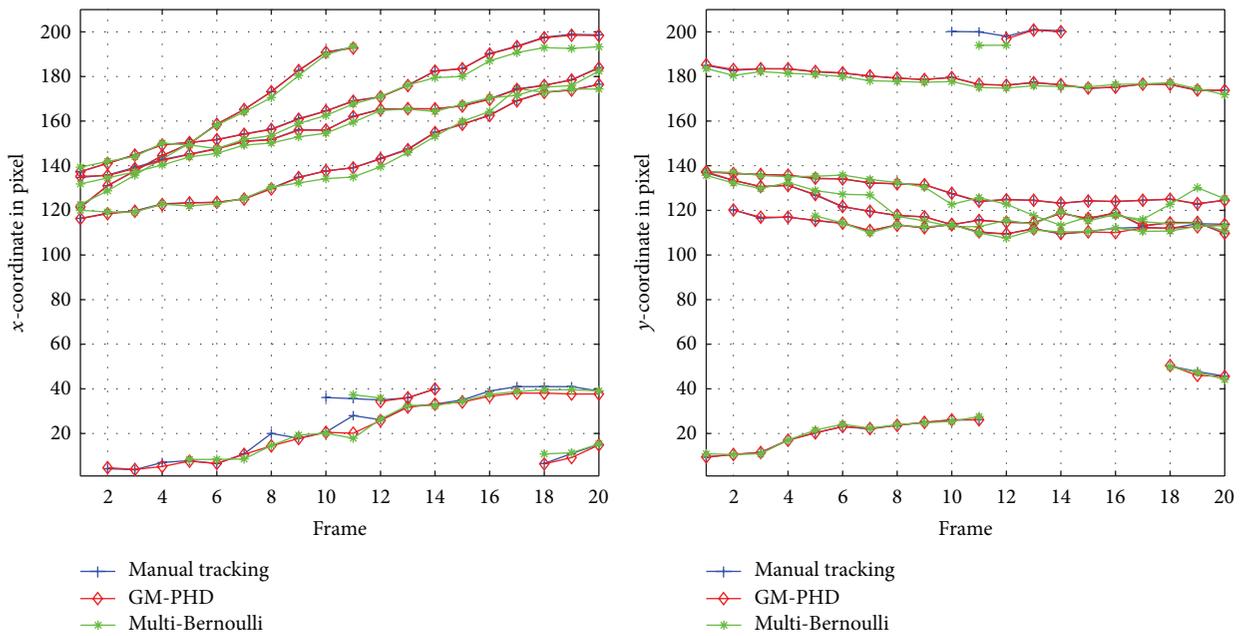


FIGURE 9: The corresponding positions estimate of each cell in  $x$  and  $y$  directions per frame using different methods.

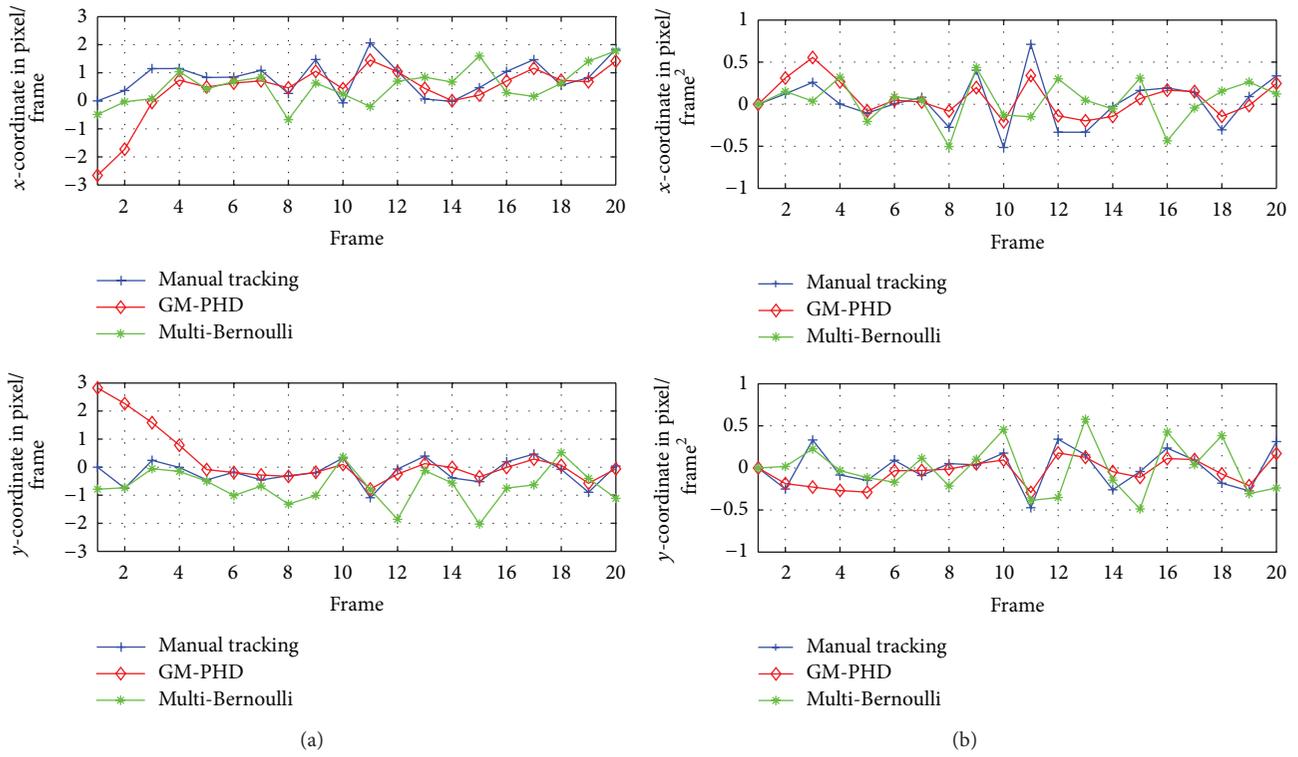


FIGURE 10: Cell instant velocity (a) and acceleration (b) estimates (cell 4 in Figure 7).

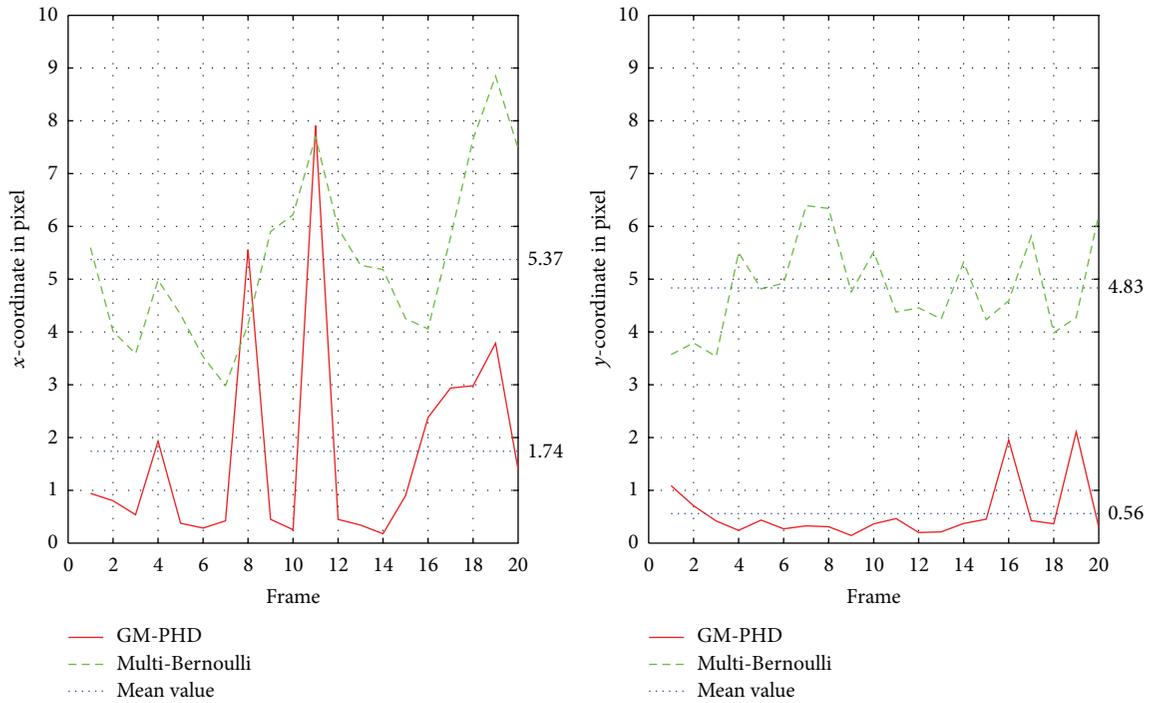


FIGURE 11: RMSE in cell position estimations using different methods.

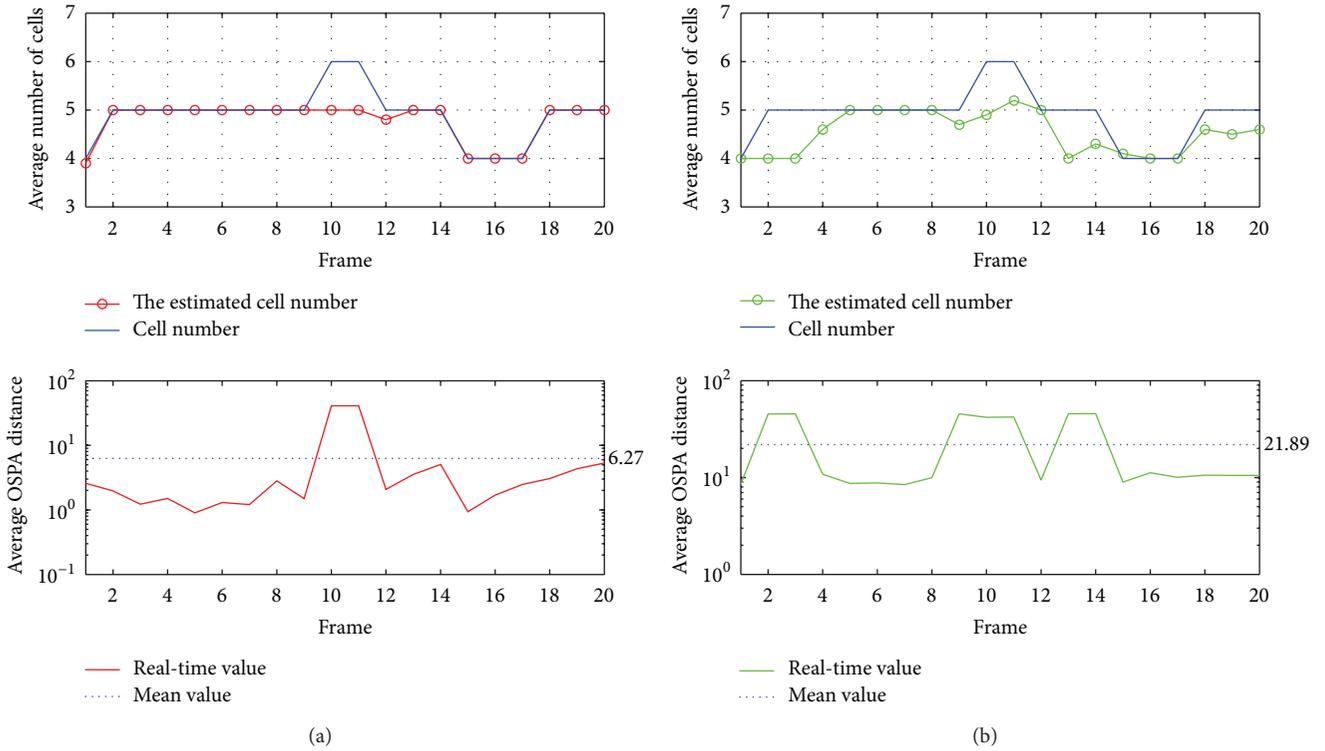


FIGURE 12: Cell number estimate and OSPA distance of two methods. ((a) The GM-PHD filter and (b) the multi-Bernoulli filter.)

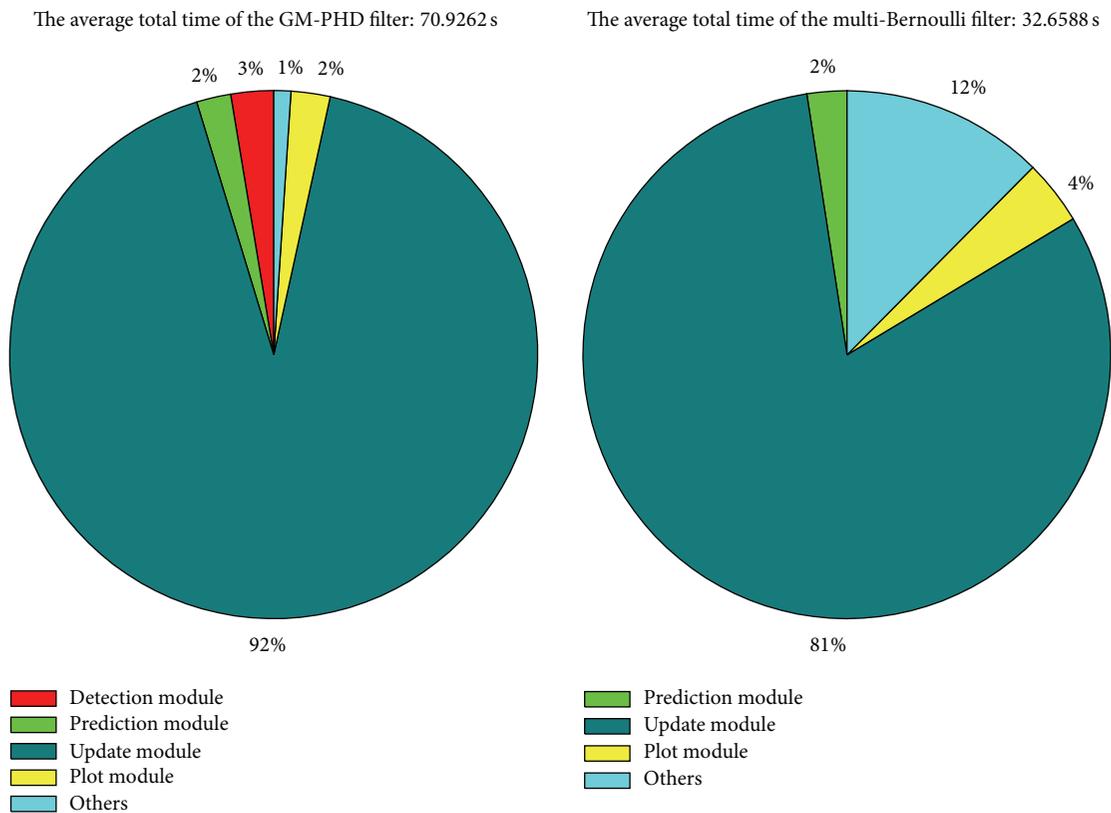


FIGURE 13: The average total time of treatment to different methods (10 independent runs).

**Read** image  $\mathbf{y}_k$ .

**Given** the posterior density  $\{\omega_{k-1}^{(i)}, m_{k-1}^{(i)}, P_{k-1}^{(i)}\}_{i=1}^{J_{k-1}}$ , the birth density  $\{\omega_{\gamma,k}^{(i)}, m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}\}_{i=1}^{J_\gamma}$ , and the cell measurement set  $\mathbf{Z}_k$  calculated by detection module.

**Step 1.** Prediction for birth cells

$i = 0$ .

for  $j = 1, \dots, J_{\gamma,k}$

$i := i + 1$ .

$\omega_{k|k-1}^{(j)} = \omega_{\gamma,k}^{(j)}, m_{k|k-1}^{(j)} = m_{\gamma,k}^{(j)}, P_{k|k-1}^{(j)} = P_{\gamma,k}^{(j)}$ .

end

**Step 2.** Prediction for existing cells

for  $j = 1, \dots, J_{k-1}$

$i := i + 1$

$\omega_{k|k-1}^{(j)} = P_{S,k} \omega_{k-1}^{(j)}, m_{k|k-1}^{(j)} = F_{k-1} m_{k-1}^{(j)}, P_{k|k-1}^{(j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T$ .

end

$J_{k|k-1} = i$ .

**Step 3.** Construction of update module

for  $j = 1, \dots, J_{k|k-1}$

$\eta_{k|k-1}^{(j)} = H_k m_{k|k-1}^{(j)}, S_k^{(j)} = R_k + H_k P_{k|k-1}^{(j)} H_k^T$ ,

$K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T [S_k^{(j)}]^{-1}, P_{k|k-1}^{(j)} = [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}$ .

end

**Step 4.** Update module

for  $j = 1, \dots, J_{k|k-1}$

$\omega_k^{(j)} = (1 - P_{D,k}) \omega_{k|k-1}^{(j)}, m_k^{(j)} = m_{k|k-1}^{(j)}, P_k^{(j)} = P_{k|k-1}^{(j)}$ .

end

$l := 0$ .

for each  $\mathbf{z} \in \mathbf{Z}_k$

$l := l + 1$ .

for  $j = 1, \dots, J_{k|k-1}$

$\omega_k^{(U_{k|k-1}+j)} = P_{D,k} \omega_{k|k-1}^{(j)} N(\mathbf{z}; \eta_{k|k-1}^{(j)}, S_k^{(j)}), m_k^{(U_{k|k-1}+j)} = m_{k|k-1}^{(j)} + K_k^{(j)} (\mathbf{z} - \eta_{k|k-1}^{(j)})$ ,

$P_k^{(U_{k|k-1}+j)} = P_{k|k-1}^{(j)}$ .

end

$\omega_k^{(U_{k|k-1}+j)} := \frac{\omega_k^{(U_{k|k-1}+j)}}{\kappa_k(\mathbf{z}) + \sum_{i=1}^{J_{k|k-1}} \omega_k^{(U_{k|k-1}+i)}}$ , for  $j = 1, \dots, J_{k|k-1}$ .

end

**Step 5.** Pruning and merging module

**Output**  $\{\omega_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_k}$ .

ALGORITHM 1

TABLE 2: Performance comparison results.

Methods	FAR (%)	FNR (%)	LSR (%)	LTR (%)
GM-PHD	0	2.58	0	0
Multi-Bernoulli	0.10	5.26	2.00	1.14

32.6588 s, respectively, and shows that the spending time of the GM-PHD filter is more than twice the spending time of the multi-Bernoulli filter.

## 5. Conclusions

In this paper, we investigate the multi-cell tracking performances using two RFS-based filtering methods, that is, Gaussian mixture probability hypothesis density filter and multi-Bernoulli filter. Numerous performance comparisons

between the two filtering methods have been carried on. According to the comparisons of our proposed performance measures, the GM-PHD filtering method has higher accuracy than the multi-Bernoulli filtering method. Specifically, the GM-PHD filter outperforms the multi-Bernoulli filter in terms of RMSE and OSPA distance, improved by 463.4% and 174.8%, respectively. Furthermore, the FAR and FNR (detection error) and LSR and LTR (tracking error) using the GM-PHD filtering method are reduced by 1.33%, 2.77%, 4.50% and 0.82% (averaged over two cases), respectively. It is noteworthy that the multi-Bernoulli filtering method has an advantage in the processing time, especially in Case 2 where the running speed is more than twice as fast as the GM-PHD. Specifically, for the GM-PHD filtering method, the average total time of Cases 1 and 2 is 38.6118 s and 70.9262 s, respectively, while for the multi-Bernoulli filtering method they are 30.0957 s and 32.6588 s, respectively. The main cause

**Read image**  $\mathbf{y}_k$

**Given** the posterior density  $\left\{ r_{k-1}^{(i)}, \left\{ w_{k-1}^{(i,j)}, \mathbf{x}_{k-1}^{(i,j)} \right\}_{j=1}^{L_{k-1}^{(i)}} \right\}_{i=1}^{M_{k-1}}$ , the birth density

$\left\{ r_{\Gamma,k}^{(i)}, \left\{ w_{\Gamma,k}^{(i,j)}, \mathbf{x}_{\Gamma,k}^{(i,j)} \right\}_{j=1}^{L_{\Gamma,k}^{(i)}} \right\}_{i=1}^{M_{\Gamma,k}}$ ,  $q_k^{(i)}(\cdot | \mathbf{x}_{k-1}, \mathbf{y}_k)$  and  $b_k^{(i)}(\cdot | \mathbf{y}_k)$  which are the densities for the particles corresponding to existing and birth cells respectively, and the cell template  $\mathbf{t}_k$ .

**Step 1. Prediction for birth cells**

$i = 0$ .

for  $j = 1, \dots, M_{\Gamma,k}$

$i := i + 1, r_{k|k-1}^{(i)} = r_{\Gamma,k}^{(j)}$ .

for  $l = 1, \dots, L_{\Gamma,k}^j$

$$\mathbf{x}_{k|k-1}^{(i,l)} \sim b_k^{(j)}(\cdot | \mathbf{y}_k), w_{k|k-1}^{(i,l)} = \frac{p_{\Gamma,k}(\mathbf{x}_{\Gamma,k}^{(j,l)})}{b_k^{(j)}(\mathbf{x}_{\Gamma,k}^{(j,l)} | \mathbf{y}_k)}.$$

end

$$w_{k|k-1}^{(i,l)} = \frac{w_{k|k-1}^{(i,l)}}{\sum_{l=1}^{r_{\Gamma,k}^{(j)}} w_{k|k-1}^{(i,l)}}, \text{ for } l = 1, \dots, L_{\Gamma,k}^j.$$

end

**Step 2. Prediction for existing cells**

for  $j = 1, \dots, M_{k-1}$

$$i := i + 1, r_{k|k-1}^{(i)} = r_{k-1}^{(j)} \sum_{l=1}^{L_{k-1}^{(j)}} w_{k-1}^{(j,l)} p_{S,k}(\mathbf{x}_{k-1}^{(j,l)}).$$

for  $l = 1, \dots, L_{k-1}^j$

$$\mathbf{x}_{k|k-1}^{(i,l)} \sim q_k^{(j)}(\cdot | \mathbf{x}_{k-1}^{(j,l)}, \mathbf{y}_k),$$

$$w_{k|k-1}^{(i,l)} = \frac{\left[ w_{k-1}^{(j,l)} f_{k|k-1}(\mathbf{x}_{k|k-1}^{(i,l)} | \mathbf{x}_{k-1}^{(j,l)}) p_{S,k}(\mathbf{x}_{k-1}^{(j,l)}) \right]}{q_k^{(j)}(\mathbf{x}_{k|k-1}^{(i,l)} | \mathbf{x}_{k-1}^{(j,l)}, \mathbf{y}_k)}.$$

end

$$w_{k|k-1}^{(i,l)} = \frac{w_{k|k-1}^{(i,l)}}{\sum_{l=1}^{r_{k-1}^{(j)}} w_{k|k-1}^{(i,l)}}, \text{ for } l = 1, \dots, L_{k-1}^j.$$

end

$M_{k|k-1} = i$ .

**Step 3. Update module**

for  $j = 1, \dots, M_{k|k-1}$

$$\rho_k^{(j)} = \sum_{l=1}^{L_{k|k-1}^{(j)}} w_{k|k-1}^{(j,l)} g_k(\mathbf{y}_k | \mathbf{x}_{k|k-1}^{(j,l)}), r_k^{(j)} = \frac{r_{k|k-1}^{(j)} \rho_k^{(j)}}{(1 - r_{k|k-1}^{(j)}) + r_{k|k-1}^{(j)} \rho_k^{(j)}}.$$

for  $l = 1, \dots, L_{k|k-1}^j$

$$\mathbf{x}_k^{(j,l)} = \mathbf{x}_{k|k-1}^{(j,l)}, w_k^{(j,l)} = \frac{w_{k|k-1}^{(j,l)} g_k(\mathbf{y}_k | \mathbf{x}_{k|k-1}^{(j,l)})}{\rho_k^{(j)}}$$

end

end

**Step 4. Resampling, pruning and merging module.**

**Output**  $\left\{ r_k^{(j)}, \left\{ w_k^{(i,j)}, \mathbf{x}_k^{(i,j)} \right\}_{j=1}^{L_k} \right\}_{i=1}^{M_k}$ .

ALGORITHM 2

is that the number of cells in Case 2 (up to 7 cells) is bigger than Case 1 (up to 3 cells); thus it needs to spend more time in the update module. However, it is noted that the multi-Bernoulli filter has noticeably lower processing time in the case of tracking numerous cells. Finally, factors affecting the performance of the GM-PHD filter include detection module (affected the tracking accuracy) and cell number

(affected the processing time), but the main factor affecting the performance of the multi-Bernoulli filter is the design of the likelihood function. section\*Appendices

## A. Pseudocode for the GM-PHD Filter

See Algorithm 1.

## B. Sequential Monte Carlo Implementation for the Multi-Bernoulli Filter

See Algorithm 2.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

This work is supported by National Natural Science Foundation of China (no. 61273312) and National Natural Science Foundation of Jiangsu province (no. BK2010261).

### References

- [1] Q. Wen, J. Gao, and K. Luby-Phelps, "Multiple interacting sub-cellular structure tracking by sequential Monte Carlo method," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM '07)*, pp. 437–442, November 2007.
- [2] I. Smal, W. Niessen, and E. Meijering, "A new detection scheme for multiple object tracking in fluorescence microscopy by joint probabilistic data association filtering," in *Proceedings of the 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI '08)*, pp. 264–267, Paris, France, May 2008.
- [3] O. Debeir, P. V. Ham, R. Kiss, and C. Decaestecker, "Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes," *IEEE Transactions on Medical Imaging*, vol. 24, no. 6, pp. 697–711, 2005.
- [4] X. Yang, H. Li, and X. Zhou, "Nuclei segmentation using marker-controlled watershed, tracking using mean-shift, and Kalman filter in time-lapse microscopy," *IEEE Transactions on Circuits and Systems I*, vol. 53, no. 11, pp. 2405–2414, 2006.
- [5] C. Zimmer and J.-C. Olivo-Marin, "Coupled parametric active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1838–1842, 2005.
- [6] N. Ray, S. T. Acton, and K. Ley, "Tracking leukocytes in vivo with shape and size constrained active contours," *IEEE Transactions on Medical Imaging*, vol. 21, no. 10, pp. 1222–1235, 2002.
- [7] D. P. Mukherjee, N. Ray, and S. T. Acton, "Level set analysis for leukocyte detection and tracking," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 562–572, 2004.
- [8] R. Hoseinnezhad, B.-N. Vo, B.-T. Vo, and D. Suter, "Visual tracking of numerous targets via multi-Bernoulli filtering of image data," *Pattern Recognition*, vol. 45, no. 10, p. 3625, 2012.
- [9] B.-N. Vo, B.-T. Vo, N.-T. Pham, and D. Suter, "Joint detection and estimation of multiple objects from image observations," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5129–5141, 2010.
- [10] R. P. S. Mahler, *Statistical Multisource Multi-Target Information Fusion*, Artech House, Norwood, Mass, USA, 2007.
- [11] B. Ristic, D. Clark, and B.-N. Vo, "Improved SMC implementation of the PHD filter," in *Proceedings of the 13th Conference on Information Fusion (Fusion '10)*, pp. 1–8, July 2010.
- [12] C. Ouyang, H.-B. Ji, and Z.-Q. Guo, "Extensions of the SMC-PHD filters for jump Markov systems," *Signal Processing*, vol. 92, no. 6, pp. 1422–1430, 2012.
- [13] R. R. Juang, A. Levchenko, and P. Burlina, "Tracking cell motion using GM-PHD," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI '09)*, pp. 1154–1157, July 2009.
- [14] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.
- [15] Y.-D. Wang, J.-K. Wu, A. A. Kassim, and W. Huang, "Data-driven probability hypothesis density filter for visual tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1085–1095, 2008.
- [16] E. Maggio, M. Taj, and A. Cavallaro, "Efficient multitarget visual tracking using random finite sets," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1016–1027, 2008.
- [17] E. Pollard, A. Plyer, B. Pannetier, F. Champagnat, and G. Le Besnerais, "GM-PHD filters for multi-object tracking in uncalibrated aerial videos," in *Proceedings of the 12th International Conference on Information Fusion (FUSION '09)*, pp. 1171–1178, July 2009.
- [18] T. Wood, C. Yates, D. Wilkinson, and G. Rosser, "Simplified multi-target tracking using the PHD filter for microscopic video data," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 5, pp. 702–713, 2012.
- [19] S. H. Rezatofighi, S. Gould, B.-N. Vo, K. Mele, W. E. Hughes, and R. Hartley, "A multiple model probability hypothesis density tracker for time-lapse cell microscopy sequences," *Information Processing in Medical Imaging*, vol. 7917, pp. 110–122, 2013.
- [20] R. P. S. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.
- [21] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer, 2001.
- [22] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 51–65, 2005.
- [23] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

