Machine Vision and Applications (2018) 29:1191–1208
https://doi.org/10.1007/s00138-018-0963-6

ORIGINAL PAPER

Visual tracking of resident space objects via an RFS-based multi-Bernoulli track-before-detect method

Mohammadreza Javanmardi[1] · Xiaojun Qi[1]

Received: 20 December 2017 / Revised: 11 June 2018 / Accepted: 27 June 2018 / Published online: 2 August 2018

Abstract
In this paper, we propose a fast and reliable track-before-detect approach to simultaneously detect, track, and identify an unknown and variable number of resident space objects (RSOs) without any prior information and any explicit detection, which leads to better space domain awareness. Specifically, we use the point spread function concept to propose a separable likelihood function as the observation model in the random finite set-based multi-Bernoulli filtering framework. This framework clearly distinguishes RSOs from any counterfeit objects and detects and tracks them immediately after their respective appearance in background cluttered telescope imagery data. The extensive experimental results on the TAOS dataset demonstrate the robustness of the proposed method in detecting and tracking RSOs with the average optimal subpattern assignment localization error less than 2 pixels in image sequences with the signal to noise ratio as low as 9 dB and under the conditions of varying illumination and occlusion.

Keywords Space domain awareness · Resident space objects · Point spread function · Multi-target tracking · Multi-Bernoulli filtering · Random finite set

1 Introduction

Reliable tracking of Resident Space objects (RSOs) in electro-optical images has been perceived as one of the most interesting research topics in space surveillance. RSOs are small objects in the space circling around the earth and are often referred to as unresolved objects because they do not visibly exhibit any physical characteristics. Several methods have been introduced in recent years to detect and track a single RSO in electro-optical image sequences under different challenges such as low signal to noise ratio (SNR) of targets, occlusion, and background clutters.

In general, RSO tracking approaches can be classified into two categories: the detect-before-track (DBT) and the track-before-detect (TBD). The DBT methods require a specific detection process to initialize the starting position of the tracked target before tracking takes place. For example, Schildknecht et al. [24] consider a group of pixels above a pre-defined threshold as the regions of interest (ROIs) for the tracking task. Yanagisawa et al. [33] incorporate an array of optical sensors as a ground-based observation system for monitoring laser electro-optical (LEO) objects. They then employ a line-identifying technique followed by thresholding and shape-based analysis to identify ROIs. Koblick et al. [11] apply the phase congruency-based segmentation method to detect RSOs. Flewelling and Sease [4] combine the Harris corner detector and the phase congruency edge detector to discriminate the RSOs from streaking stars. All these approaches have successfully tracked RSOs by compressing the information in each frame to a small set of ROIs. However, they may become computational bottleneck as the number of images and hypothesized candidates in a track drastically increase. Furthermore, they may incur an inherent loss of information due to Boolean decisions.

Consequently, the TBD approaches have been proposed to take intensity information from a track as the input and simultaneously output the estimated cardinality distribution along with the joint distribution of the target states conditioned upon the cardinality. The implementation can be a brute-force search for the motion of RSOs in the image plane via image stacking [34]. More sophisticated approaches

✉ Mohammadreza Javanmardi
Javanmardi@aggiemail.usu.edu

Xiaojun Qi
Xiaojun.Qi@usu.edu

[1] Department of Computer Science, Utah State University, Logan, UT 84322-4205, USA

involve a Bayesian estimator where the measurement likelihood function of RSOs may be determined by a genetic algorithm [26] or a bank of templates [17] evolved from partial a priori state information. More recent method [5] applies the Bernoulli particle filter, which is implemented by Sequential Monte Carlo (SMC) method with a likelihood function, to simultaneously detect and track one RSO in images of low SNR. Despite the reliable performance of these methods in tracking a single RSO, developing a robust method to detect, track, and identify an unknown and variable number of RSOs in image sequences has been remained unsolved.

This paper proposes a fast and reliable multi-RSO tracking framework by employing the multi-target multi-Bernoulli (MeMBer) filter [8,15,28] to simultaneously detect, identify, and track an unknown and variable number of RSOs immediately after their respective first appearance in low-SNR CCD image sequences. The proposed method does not need any prior information and any explicit detection. The MeMBer filter is derived from the Random Finite Set (RFS) and is a tractable solution to the multi-target Bayes filter, which propagates the multi-Bernoulli parameters of the multi-target posterior distribution forward in time. To the best of our knowledge, this paper is the first attempt to apply the multi-Bernoulli RFS framework in space domain awareness (SDA) [10] to track multiple RSOs in background cluttered and low-SNR telescope imagery data. Our contributions are: (1) introducing a separable likelihood, which is able to make a clear distinction between RSOs and any counterfeit objects such as streaking stars, bright stars, and any cosmic rays signals in low-SNR frame sequences. (2) modeling both multi-target states and observations in multi-Bernoulli RFS framework for efficiently refining and rejecting the target measurement likelihood function. (3) employing the multi-Bernoulli filtering approach to propagate the multi-Bernoulli RFS parameters forward in time for automatically and quickly detecting and tracking an unknown and variable number RSOs in unresolved imagery. (4) designing a labeling and occlusion management strategy to utilize the history of the presence or absence of targets and their associated applicable velocity information for reliably and accurately identifying each RSO with its unique label even when it is occluded by any other space objects or background clutter. (5) developing an adaptive likelihood thresholding method to utilize the history information for removing the chaotic noise. The extensive experimental results demonstrate that the proposed approach is able to robustly detect, track, and identify multiple RSOs with their own identification IDs in clutters and in low-SNR telescope imagery data.

The rest of the paper is organized as follows: Sect. 2 reviews the current trend of DBT and TBD approaches in objects tracking. Section 3 explains four challenges associated with detecting and tracking RSOs and defines the proposed likelihood function to seamlessly address these challenges. Section 4 describes the MeMBer filtering method that is incorporated in the proposed tracker. Section 5 presents the implementation details and the labeling and occlusion management. Section 6 shows the experimental results on various tracks captured by the TAOS project and evaluates the performance of the proposed tracking approach in terms of the average position estimation error in pixels and the optimal subpattern assignment (OSPA) distance together with its localization and cardinality errors in pixels. Section 7 draws the conclusions and presents the directions of future work.

2 Background

In recent years, researchers have been actively studying various DBT and TBD approaches to solve the problem of jointly estimating the number of objects and their states from image observation. Conventional DBT approaches [2,12,22,35] attain good performance to track objects with large SNR and decent size. However, they fail to track objects with low SNRs and tend to detect a large number of outliers or completely miss the target. Unlike the DBT approaches that declare the presence of targets at each scan before tracking, the TBD approaches do not need any explicit detection process to achieve superior detection performance by jointly processing multiple consecutive scans [1,9,18,21]. For instance, Buzzi et al. [1] derive a generalized likelihood ratio test (GLRT) as an extension to the non-Bayesian approach to achieve better estimation and tracking accuracy with a lower complexity in radar images. Prez et al. [21] and Nummiaro et al. [18] individually propose a color-based SMC framework and an adaptive color-based particle filtering framework to define a likelihood function, which compares the color information of candidates and the target model to address the shading and illumination effects. These TBD methods are simple and robust in solving a wide range of tracking problems. However, a high-dimensional SMC approximation of integrals is required to model multi-target Bayes filter and the consistent maintenance of the multimodality in the target distribution is deficient due to insufficient measurement and ambiguity of multiple objects and clutters.

Several Bayesian data association approaches have been proposed to address the aforementioned dimensionality and multimodality issues [3,19,27]. One example is the work by Czyz et al. [3]. They present a hybrid-valued (continuous-discrete) sequential state estimation algorithm to detect and track multiple targets of similar color by assuming that the background is of a sufficiently different color than the tracked objects. They include a discrete variable representing the number of targets in their states to simultaneously estimate the states and cardinality. All these methods address the dimensionality and multimodality issues to achieve more robust tracking performance. However, the required associations between targets and measurements may result in difficulty in its mathematical formulations and lead to combinatorial growth in the number of hypotheses.

To address any target and measurement association-related issues, the RFSs have been proposed. Mahlers finite set statistics (FISST), an intuitive and natural representation of multiple target states and measurements, provides practical mathematical tools for dealing with the RFSs. Mahler [15] proposes probability hypothesis density (PHD), the first-order moment of multi-target posterior, as a practical alternative to the multi-target optimal filter. To estimate the cardinality more efficiently, Mahler [13] generalizes the PHD recursion by relaxing the first-order assumption and derives a closed-form cardinalized PHD (CPHD) filter. Vo et al. [30] propose a closed-form solution to CPHD under linear Gaussian assumptions on the target dynamics and birth process. In the seminal work [15], Mahler also introduces multi-Bernoulli filter to approximate the multi-target posterior density and propagate the related parameters in time. The idea of multi-Bernoulli filter is later employed in [28] to propose an RFS TBD model and a tractable filter to track multiple non-overlapping targets with low SNR in radar images using a separable likelihood function. Hosseinnezhad et al. [8] derive a separable multi-target color-based likelihood function in the multi-Bernoulli framework using camera image observations. They further employ kernel density estimation [7] to update the background and subtract the learned background from the original frames. It should be noted that the likelihood function in these methods has to be separable for the individual targets with the assumption that each target independently influences the image observation. This assumption is reasonable when the objects do not overlap with each other in all the frames. As one of the most recent approaches, a tractable generalized labeled multi-Bernoulli (GLMB) is proposed by Papi et al. [20] to match the cardinality distribution and the first moment of the labeled multi-object distribution. This method is applicable to not only separable likelihood cases but also more general cases when multi-target likelihood is not separable.

Unlike current multi-Bernoulli-based tracking methods [7,8,28], which are applicable for detecting and tracking targets in radar or camera images, the proposed method can be applied on CCD imagery data to track RSOs. It seamlessly incorporates the following special properties of CCD images into the likelihood function: (1) target regions are significantly blurry; (2) the tracked RSO objects are unresolved small objects that do not visibly exhibit any physical characteristics; (3) there are many spurious moving objects similar to RSO objects; and (4) the images have cluttered background in noisy environment.

3 Visual likelihood function for fainted objects

Providing robust small-sized target detection and tracking has significant applications in medical, military, SDA, and so on. In SDA, locating the moving RSO in CCD imagery data, where RSOs normally have a few pixels in size and have low contrast compared to the background, is the first step to make crucial decisions to maintain the peace of the space. However, there are several challenges in detecting and tracking RSOs in the telescope imagery data: (1) low SNR; (2) blurring effect of the imaging system; (3) similarity between RSOs and the bright stars or any signals caused by cosmic rays; and (4) similarity between RSOs and streaking stars.

In this paper, we propose a novel TBD tracking method, which employs the multi-Bernoulli filtering framework to model both multi-target states and observations, to address the four aforementioned challenges to automatically detect and track multiple RSOs in the noisy cluttered environment. In the next subsections, we explain the four challenges in detail and describe the proposed likelihood function for fainted objects (e.g., RSOs) to address all the four challenges.

3.1 Four challenges in tracking RSOs

We explain each of the four challenges in detail:

1. *Low SNR* One of the most difficult tasks in visual tracking and object recognition is to detect and track the spot (small) targets, especially when the SNR is low and the targets are fainted in a sequence of frames. The target can be correctly detected if we can estimate the image background and eliminate the noise in the image sequence. However, it is impossible to accurately estimate the background and noise in each image frame due to low SNR values in real applications. Furthermore, the RSO can be as small as two or three pixels in size and have the low SNR value of 9 dB as in our telescope imagery datasets.
2. *Blurring effect of the imaging system* The light from point source objects ideally should be at most one pixel in the corresponding image due to their nature. However, point sources, when captured by electro-optical (EO) sensors, appear spread out over an area due to diffraction, aberration, atmospheric turbulence, and other imperfections in the light path [10]. As a result, their light actually spreads out in an area more than one pixel, which is collectively characterized through PSF. Since the signal of point source objects can be considered as a Dirac delta function $\delta(x)$, the PSF can be defined as an impulse response of the imaging system. By having the PSF of an imaging system, the blurred signal of space objects in an image region is generated by [10]:

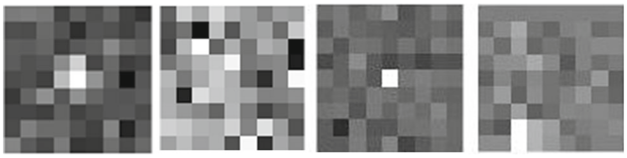


Fig. 1 Illustration of sample regions of (left to right): RSO; cluttered background; cosmic rays/bright star; and streaking stars

$$y(i,j) = \sum_{m=1}^{M}\sum_{n=1}^{N} S(m.n)h(i-m, j-n) \tag{1}$$

where $y(i,j)$ is the blurred image whose size is $M \times N$, h is the PSF (i.e., the impulse response), and S is the signal of the space object.

3. *Similarity between RSOs and the bright stars or any signals caused by cosmic rays* The cosmic rays are energetic and subatomic particles that hit the CCD to produce sharp spurious bright signals that are similar to the RSOs and bright stars. Stars are also naturally source point inputs that can have a similar signal as the RSOs.
4. *Similarity between RSOs and streaking stars* There are other undesirable signals in CCD images due to streaking stars, where each individual star exhibits the similar characteristics as a RSO.

Figure 1 shows the sample regions for a RSO, a cluttered background without any specific object, bright star or cosmic ray with a peaked signal, and the streaking star with an elongated signal. It clearly shows that the counterfeit signals produce the bright pixels, which are similar to the RSO pixels. Furthermore, it clearly presents the blurring effects in the CCD images due to the PSF of the imaging system.

3.2 Likelihood measurement

To provide a robust tracking algorithm for multiple RSOs, we propose to incorporate a PSF-based separable likelihood function in the multi-Bernoulli filtering framework to simultaneously detect and track RSOs without explicit detection. The innovative concept behind the proposed likelihood function is to make a clear distinction among the regions of the candidates (e.g., individual RSOs and counterfeit objects) by addressing four aforementioned challenges. Moreover, we derive a separable likelihood function for multiple RSOs to update the posterior density of the states of RSOs based on noisy and background cluttered images and make a proper distinction between the states of RSOs and the states of spurious objects such as bright stars, the cosmic rays signals, and streaking stars.

Since the CCD imagery data are blurry in nature, reducing the blurring effect is crucial in constructing an effective likelihood function. Suppose that we have the states of two target candidates $\mathbf{x}_1$ and $\mathbf{x}_2$ ($\mathbf{x}_i$ represents a four-dimensional state vector containing the location and velocity in x and y coordinates) in frame k, where $\mathbf{x}_1$ represents the state of a RSO and $\mathbf{x}_2$ represents the state of a non-RSO pixel in the noisy background. We define $T(\mathbf{x}_1)$ and $T(\mathbf{x}_2)$ as regions in the original frame $\mathbf{Y}_k$ for the corresponding candidates $\mathbf{x}_1$ and $\mathbf{x}_2$, where $T(\cdot)$ is a 9×9 window that is centered at the coordinates of target candidates positions. We use the PSF [5] to reduce the blurring effect in $T(\cdot)$ regions for the state of any target candidate in frame k by computing h value for each pixel m_i as follows:

$$h_k(m_i) = \frac{I}{2\pi\sigma_h^2} e^{\frac{-[(m_{ix}-c_x)^2+(m_{iy}-c_y)^2]}{2\sigma_h^2}} \tag{2}$$

where m_i represents a pixel in $T(\cdot)$ at location of (m_{ix}, m_{iy}), (c_x, c_y) is the location of state $\mathbf{x}$ in frame $\mathbf{Y}_k$ (i.e., the center of $T(\cdot)$), I is the predefined source intensity, and σ_h^2 is the blurring factor. This computation ensures that the pixels that are farther away from the center of the $T(\cdot)$ region have smaller h values. The intensity of each pixel in $T(\mathbf{x})$ in the context of the observation (frame) $\mathbf{Y}_k$ is updated by g_{ratio} value as follows:

$$g_{\text{ratio}}(\mathbf{Y}_k; \mathbf{x}) = e^{\frac{[\mu_0 - h_k(m_i)][\mu_0 - 2y_k(m_i) + h_k(m_i)]}{2\sigma_0^2}} \tag{3}$$

where $y_k(m_i)$ is the original pixel intensity at the location of pixel m_i in frame $\mathbf{Y}_k$, μ_0 is the noise mean, and σ_0^2 is the noise variance. This computation ensures that the minimum value of g_{ratio} is 1.

Figure 2a through Fig. 2d, respectively, presents 9×9 windows (shown as the leftmost images) in the original frame around the locations of four states, namely $\mathbf{x}_1$ (state of a RSO), $\mathbf{x}_2$ (state of a non-RSO), $\mathbf{x}_3$ (state of a bright star), and $\mathbf{x}_4$ (state of a streaking star), their processed $T(\cdot)$ regions (shown as the middle images) after applying (2) and (3), and their updated $T(\cdot)$ regions (shown as the rightmost images) after applying (4), a statistics-based analysis that will be dis-

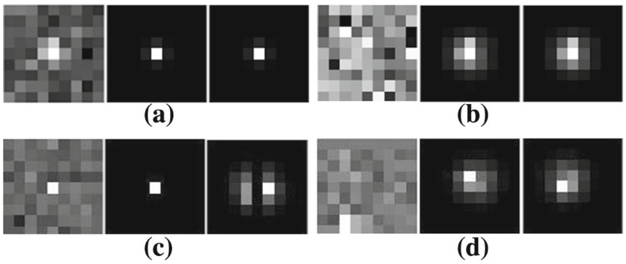


Fig. 2 Illustration of the regions around four states (shown at the left), their processed $T(\cdot)$ regions after applying (2) and (3) (shown in the middle), and their updated $T(\cdot)$ regions after applying (4) (shown at the right). Four states are for **a** RSO; **b** non-RSO pixel; **c** bright star; and **d** streaking star

cussed next. The middle images in Fig. 2a through Fig. 2d clearly show that the blurring effect of the regions of all four states is effectively reduced. For instance, $T(\mathbf{x}_1)$ region contains exactly one brightest pixel in the middle corresponding to the source point input (e.g., RSO target) whereas the remaining pixels are set to dark or close to dark. $T(\mathbf{x}_2)$ region still spreads out in a small middle region with a few bright pixels. However, it clears out the remaining pixels as dark or close to dark. As a result, the processed $T(\cdot)$ regions (i.e., reduced blurring regions) are able to make a clear distinction between the state of RSO and the state of a non-object pixel in cluttered background. However, we can see from Fig. 2a that the processed $T(\mathbf{x}_3)$ region is identical to the processed $T(\mathbf{x}_1)$ region and the processed $T(\mathbf{x}_4)$ region is also similar to the processed $T(\mathbf{x}_1)$ region, which demonstrate that the point source counterfeit signals have similar processed $T(\cdot)$ regions as the RSO signals.

To solve this problem, we employ statistics-based analysis to distinguish the counterfeit signals from the RSO signals using the updated g_{ratio} values of the $T(\cdot)$ region, which are computed based on the results from the kurtosis-based and student-t-based normality hypothesis tests. The use of these two hypothesis tests is mainly motivated from the following two observations: (1) pixels in CCD images for the RSO signals have a normal distribution compared to the pixels for the signals of the bright stars or cosmic rays, which appear to be leptokurtic (peaked) on CCD images; and (2) pixels in CCD images for the RSO signals are distributed more based on the normal distribution rather than the t-distribution compared to the streaking stars, which appear to be elongated on CCD images. Specifically, we conduct the kurtosis-based normality hypothesis test for the eight pixel intensities in the original frame along each horizontal scanline within the 9×9 $T(\cdot)$ region to distinguish RSOs from bright stars. The pixel corresponding to the maximum g_{ratio} value along each horizontal scanline is not involved in this computation. Similarly, we conduct the student-t normality hypothesis test for the same eight pixel intensities along each horizontal scanline to distinguish RSOs from streaking stars. If the kurtosis value (i.e., ku) of a horizontal scanline is larger than a pre-defined $k_{\min}$ value (e.g., 6), the signal is considered to be "peaked" due to cosmic ray or bright stars. If the student t test value (i.e., tt) of a horizontal scanline is smaller than a pre-defined $t_{\max}$ value (e.g., 0.01), the signal is considered as more spread out due to a streaking star. The final g_{ratio} for each horizontal scanline is updated as follows:

$$g_{\text{ratio}}(\mathbf{Y}_k; \mathbf{x}) = \begin{cases} \mathcal{N}[\mathbf{Y}_k; 0, \sigma_0^2] & \text{if } (tt < t_{\max} \text{ or } ku > k_{\min}) \\ g_{\text{ratio}}(\mathbf{Y}_k; \mathbf{x}) & \text{otherwise} \end{cases} \tag{4}$$

where $\mathcal{N}[\mathbf{Y}_k; 0, \sigma_0^2]$ represents an operation on each g_{ratio} value along a selected horizontal scanline so the distribution of the newly updated g_{ratio} values follows a normal distribution with the mean of 0 and the variance of σ_0^2. In other words, if either the kurtosis value or the t test value indicates the inconsistency with the expected distributions of the RSO signals, we will update the corresponding g_{ratio} values to follow a normal distribution.

The rightmost images in Fig. 2a through Fig. 2d, respectively, present the updated $T(\cdot)$ regions of four different objects, namely RSO, non-RSO, bright star, and streaking star, after applying (4). They clearly demonstrate that these updated $T(\cdot)$ regions are distinct from each other. It should be noted that this statistics-based analysis method changes the g_{ratio} values in the processed $T(\cdot)$ regions of states $\mathbf{x}_3$ and $\mathbf{x}_4$ and keeps the g_{ratio} values in the processed $T(\cdot)$ regions of states $\mathbf{x}_1$ and $\mathbf{x}_2$ intact. This further illustrates the effectiveness of using kurtosis-based and student-t-based normality hypothesis tests to distinguish the two kinds of point source counterfeit signals from the RSO signals.

The likelihood parameter (i.e., g_f) for the state of a target candidate (i.e.,$\check{\mathbf{x}}$) in an image $\mathbf{Y}$ can then be obtained by:

$$g_f(\mathbf{Y}; \check{\mathbf{x}}) = \zeta \prod_{m_i \in T(\check{\mathbf{x}})} m_i \tag{5}$$

where ζ is a normalization factor to ensure that the likelihood function integrates to one for all the observations in the observation space given a target state and m_i is the value of g_{ratio} at the location of (m_{ix}, m_{iy}) in the $T(\cdot)$ region, which is computed by (4).

Given an image $\mathbf{Y}$ in a frame sequence and a multi-target state $\check{\mathbf{X}} = [\check{\mathbf{x}}_1, \ldots, \check{\mathbf{x}}_n]$, we can derive the multi-target measurement likelihood function $g(\mathbf{Y}|\check{\mathbf{X}})$. Since the RSOs do not overlap with each other in all the image frames, we can safely assume that the signals of individual RSOs on the image observation are independently distributed and each RSO does not considerably affect other RSOs' $T(\cdot)$ regions. Moreover, the background pixels are assumed to be independently distributed based on the normal distribution with the mean of 0 and the variance of σ^2 ($\mathcal{N}(:, 0, \sigma^2)$). These two assumptions are valid since the source signal of each RSO is independent and the regions occupied by each RSO are represented as a group of pixels, which are less probable to completely slide on each other throughout the frame sequence. Following these two assumptions, the likelihood function for multi-target state is formulated as follows:

$$g(\mathbf{Y}|\check{\mathbf{X}}) = g_b(\mathbf{Y}) \prod_{i=1}^{n} g_f(\mathbf{Y}; \check{\mathbf{x}}_i) \tag{6}$$

where $g_b(\mathbf{Y}) = \prod_s \mathcal{N}(\mathbf{Y}(s), 0, \sigma^2)$ is the likelihood value of the sth background pixel in the frame $\mathbf{Y}$, g_f is the likelihood value calculated from (5) for the target $\check{\mathbf{x}}_i$ in the multi-target state $\check{\mathbf{X}}$, and n is the number of targets. This likelihood function is separable since it can be written as a product of functions $g_b(\mathbf{Y})$ and $g_f(\mathbf{Y}; \check{\mathbf{x}}_i)$, where $g_b(\mathbf{Y})$ is independent of the target states and $g_f(\mathbf{Y}; \check{\mathbf{x}}_i)$ depends only on one of the target states in the multi-target state $\check{\mathbf{X}}$ and is independent of the background pixels. This separable form of the likelihood function will be used later in Sect. 4 to update the multi-Bernoulli parameters.

4 Multi-Bernoulli filtering technique

We employ the Bayesian-based multi-target filtering technique to compute the joint estimation of the number of targets and states using observations and propagate the multi-target posterior density recursively in time. To this end, we model both multi-target states and observations as RFSs. RFSs operate on unordered finite sets and therefore are fully specified by the distribution of the number of elements in the set (cardinality) and the joint distribution of the elements conditioned upon the cardinality [6]. As such, we use Bernoulli RFS filters to explicitly account for the birth and death of multiple RSOs in some measurement arc to achieve TBD via SMC implementation.

Suppose that there are N_k targets and M_k observations at time k, the states of these targets are represented as $\check{\mathbf{x}}_{k,1}, \ldots, \check{\mathbf{x}}_{k,N_k}$ and the observations are represented as $\check{\mathbf{y}}_{k,1}, \ldots, \check{\mathbf{y}}_{k,M_k}$ in the multi-target state space $\check{\mathcal{X}}$ and the multi-target observation space $\check{\mathcal{Y}}$, respectively. The multi-target states and observations at time k can be cast as two Bernoulli RFSs [14,32]:

$$\check{\mathbf{X}}_k = \{\check{\mathbf{x}}_{k,1}, \ldots, \check{\mathbf{x}}_{k,N_k}\} \subset \check{\mathcal{X}} \tag{7a}$$

$$\mathbf{Y_k} = \{\check{\mathbf{y}}_{k,1}, \ldots, \check{\mathbf{y}}_{k,M_k}\} \subset \check{\mathcal{Y}} \tag{7b}$$

The Bernoulli RFS $\check{\mathbf{X}}_k$ can be further defined by two Bernoulli parameters: r_k and p_k, where r_k is the probability of being singleton whose only element (target) is distributed according to its probability density p_k defined on $\check{\mathcal{X}}$ (e.g., $r_k \cdot p_k(\check{\mathbf{x}})$). In addition, the Bernoulli RFS $\check{\mathbf{X}}_k$ has the probability of $1 - r_k$ to be an empty set (i.e., contains no target) and has the probability of 0 to contain two or more targets. Mathematically, the probability of the Bernoulli RFS $\check{\mathbf{X}}_k$ is defined as follows [32]:

$$\pi(\check{\mathbf{X}}_k) = \begin{cases} 1 - r_k & \text{if } \check{\mathbf{X}}_k = \emptyset \\ r_k \cdot p_k(\check{\mathbf{x}}) & \text{if } \check{\mathbf{X}}_k = \{\check{\mathbf{x}}\} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

A multi-Bernoulli RFS is defined as a union of a fixed number of independent Bernoulli RFSs such as $\check{\mathbf{X}}_k^i$ with $(r_k^{(i)}, p_k^{(i)})$ for $i = 1, \ldots, M_k$, i.e., $\{(r_k^{(i)}, p_k^{(i)})\}_{i=1}^{M_k}$. The prediction and update steps are iteratively employed to propagate the multi-Bernoulli parameters of multi-target posterior forward in time [16,28,31,32].

4.1 Prediction step

Following Mahlers notation in [6,13], if the posterior multi-target density at time $k-1$ is a multi-Bernoulli of the form $\pi_{k-1} = \{(r_{k-1}^{(i)}, p_{k-1}^{(i)})\}_{i=1}^{M_{k-1}}$, the predicted multi-target density is also a multi-Bernoulli consisting of newborn and surviving RFSs [28,32]:

$$\pi_{k|k-1} = \{(r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)})\}_{i=1}^{M_{\Gamma,k}} \cup \{(r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)})\}_{i=1}^{M_{k-1}} \tag{9}$$

where

$$r_{P,k|k-1}^{(i)} = r_{k-1}^{(i)} < p_{k-1}^{(i)}, p_{S,k} > \tag{10}$$

and

$$p_{P,k|k-1}^{(i)}(\check{\mathbf{x}}) = \frac{< f_{k|k-1}(\check{\mathbf{x}}|\cdot), p_{k-1}^{(i)} p_{S,k} >}{< p_{k-1}^{(i)}, p_{S,k} >} \tag{11}$$

Here, subscript Γ represents the newborn RFSs, subscript P represents the persistently existing RFSs, $< \alpha, \beta >= \int \alpha(x)\beta(x)d(x)$, $p_{S,k}$ is the survival probability of target at time k given the previous state, and $f_{k|k-1}(\cdot|\cdot)$ is multi-target transition density from time $k-1$ to k . The prediction of multi-Bernoulli parameters computed in (10) and (11) is used as a prior for calculation of the update step in the filtering process.

4.2 Update step

Following Corollary 3 in [28], we denote the predicated multi-target density $\pi_{k|k-1} = \{(r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)})\}_{i=1}^{M_{k|k-1}}$ at time k as a prior, where $M_{k|k-1} = M_{\Gamma,k} + M_{k-1}$. The posterior multi-target density $\pi_k = \{(r_k^{(i)}, p_k^{(i)})\}_{i=1}^{M_k}$ can be defined by using the separable likelihood function in (6) as follows:

$$r_k^{(i)} = \frac{r_{k|k-1}^{(i)} < p_{k|k-1}^{(i)}(\check{\mathbf{x}}), g_f(\check{\mathbf{x}}, \mathbf{Y}) >}{1 - r_{k|k-1}^{(i)} + r_{k|k-1}^{(i)} < p_{k|k-1}^{(i)}(\check{\mathbf{x}}), g_f(\check{\mathbf{x}}, \mathbf{Y}) >} \tag{12}$$

$$p_k^{(i)}(\check{\mathbf{x}}) = \frac{p_{k|k-1}^{(i)}(\check{\mathbf{x}}) g_f(\check{\mathbf{x}}, \mathbf{Y})}{< p_{k|k-1}^{(i)}(\check{\mathbf{x}}), g_f(\check{\mathbf{x}}, \mathbf{Y}) >} \tag{13}$$

It should be noted that the integrals in both the prediction and the update steps are intractable and cannot be solved analytically. As a result, we use the SMC method to implement this filter, whose details will be provided in Sect. 5.

5 Implementation

Implementation of the proposed tracker contains three main steps that are performed on each frame of a sequence. The first step is to compute multi-Bernoulli filtering and likelihood for generating and updating the multi-Bernoulli RFSs forward in time. The second step is to manage labeling and occlusion for labeling the identity of each target throughout the sequence and recognizing the occluded target with its known identity at the estimated location. The third step is to decide the adaptive threshold to remove some noisy pixels.

5.1 Multi-Bernoulli filtering and likelihood computation

The aim of this step is to use the SMC method to implement the prediction and update steps of multi-Bernoulli parameters to address the intractable integration issues. Suppose that the multi-Bernoulli posterior $\pi_{k-1} = \{(r_{k-1}^{(i)}, p_{k-1}^{(i)})\}_{i=1}^{M_{k-1}}$ at time $k-1$ is given for $i = 1, \ldots, M_{k-1}$ Bernoulli RFSs. Each $p_{k-1}^{(i)}$ can also be represented as a set of weighted particles $\{(\mathbf{w}_{k-1}^{(i,j)}, \check{\mathbf{x}}_{k-1}^{(i,j)})\}_{j=1}^{L_{k-1}^{(i)}}$, whose number equals to $L_{k-1}^{(i)}$. Here, the lower subscript of $\mathbf{w}_{k-1}^{(i,j)}$ and $\check{\mathbf{x}}_{k-1}^{(i,j)}$ represents time (or frame) $k-1$, the first element in the tuple of the upper subscript of $\mathbf{w}_{k-1}^{(i,j)}$ and $\check{\mathbf{x}}_{k-1}^{(i,j)}$ represents the ith index number for M_{k-1} Bernoulli RFSs, and the second element in the tuple of the upper subscript of $\mathbf{w}_{k-1}^{(i,j)}$ and $\check{\mathbf{x}}_{k-1}^{(i,j)}$ represents the jth index number for $L_{k-1}^{(i)}$ particles. In the prediction step, the multi-Bernoulli parameters, the particles, and their corresponding weights for the existing RFSs are propagated forward in time as follows[28]:

$$r_{P,k|k-1}^{(i)} = r_{k-1}^{(i)} \sum_{j=1}^{L_{k-1}^{(i)}} \mathbf{w}_{k-1}^{(i,j)} p_{S,k}(\check{\mathbf{x}}_{k-1}^{(i,j)}) \tag{14}$$

$$\check{\mathbf{x}}_{P,k|k-1}^{(i,j)} \sim f_{k|k-1}(\cdot|\check{\mathbf{x}}_{k-1}^{(i,j)}) \tag{15}$$

$$\mathbf{w}_{P,k|k-1}^{(i,j)} = \mathbf{w}_{k-1}^{(i,j)} \tag{16}$$

For the newborn RFSs $\{(r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)})\}_{i=1}^{M_{\Gamma,k}}$, multi-Bernoulli parameters are given by the birth model, the particles are propagated randomly, and their corresponding weights are initialized. Using the predicted multi-target density, $\pi_{k|k-1}$, which is the union set of predicted existing RFSs and newborn RFSs, the weights of the particles are then updated based on the likelihood value g_f [28]:

$$\mathbf{w}_k^{(i,j)} = \frac{\mathbf{w}_{k|k-1}^{(i,j)} g_f(\mathbf{Y}_k; \check{\mathbf{x}}_{k|k-1}^{(i,j)})}{Q_k^{(i)}} \tag{17}$$

where $Q_k^{(i)} = \sum_{j=1}^{L_{k|k-1}^{(i)}} \mathbf{w}_{k|k-1}^{(i,j)} g_f(\mathbf{Y}_k; \check{\mathbf{x}}_{k|k-1}^{(i,j)})$. These weights are further used to compute the updated Bernoulli parameters by:

$$r_k^{(i)} = \frac{r_{k|k-1}^{(i)} Q_k^{(i)}}{1 - r_{k|k-1}^{(i)} + r_{k|k-1}^{(i)} Q_k^{(i)}} \tag{18}$$

$$p_k^{(i)} = \sum_{j=1}^{L_{k|k-1}^{(i)}} \mathbf{w}_k^{(i,j)} \delta_{\check{\mathbf{x}}_{k|k-1}^{(i,j)}}(\check{\mathbf{x}}) \tag{19}$$

The updated particles in each RFS are resampled based on their corresponding weights and replicated proportionally.

Since RSOs can appear in any frame of the sequence, we consider a constant birth model for generating newborn Bernoulli RFSs in all the frames to detect the RSOs immediately after they infiltrate. At any frame k, a multi-Bernoulli RFS, which contains four Bernoulli RFSs with their individual probability of existence being 0.25, is generated (i.e.,$\{(r_{\Gamma,k}^{(i)}, p_{\Gamma,k}^{(i)})\}_{i=1}^{4}$ with $r_{\Gamma,k}^{(i)} = 0.25$). Based on SMC implementation of the multi-Bernoulli filter, $L_{\max}$ number of particles for each of the Bernoulli RFSs is initiated. The number of particles is restricted in the range of $[L_{\min}, L_{\max}]$ as suggested in [7,8] to avoid the high computational cost. Each particle is a four-dimensional vector containing the position of the candidate state (x, y) and velocities of the particle in x and y directions (v_x, v_y). The locations of particles in each RFS are uniformly distributed in any quarter of the image plane, and their velocities in x and y directions are randomly initiated in the range of $[-2, 2]$ and $[-7, 7]$, respectively. Since some Bernoulli RFSs may persistently exist till frame k, we cast them into an existing multi-Bernoulli set $\{(r_{P,k|k-1}^{(i)}, p_{P,k|k-1}^{(i)})\}_{i=1}^{M_{k-1}}$. The particles of the birth and existing multi-Bernoulli RFSs are integrated and processed in the likelihood function to calculate $g_f(\cdot; \check{\mathbf{x}}_{k|k-1}^{(i,j)})$ using (5). The likelihood parameter is then used to update the existence probability of each RFS using (18) and (19). To limit the growing number of Bernoulli RFSs, we keep the updated Bernoulli RFSs $\pi_k = \{(r_k^{(i)}, p_k^{(i)})\}_{i=1}^{M_k}$, whose $r_k^{(i)}$ is larger than a pre-defined threshold. Since an existing singleton RFS and a newborn RFS might represent the same target in a frame, we merge them into one RFS if the Euclidean distance between the average position of particles in the singleton RFS and the average position of particles in the newborn RFS is less than a pre-defined threshold (e.g., ten pixels). The merged RFS contains the particles of high probability from both RFSs, where the number of particles in the merged RFS is the minimum of $L_{\max}$ and the total number of the particles in both sets. The r value of the merged set is the minimum of 1 and the summation of r values of both set. The p value of the merged set is the summation of p values of all the particles in the merged set.

5.2 Labeling and occlusion management

The aim of this step is to accurately label the identified target as either one of the existing targets or a new target regardless of occlusion. In the case of occlusion (two RSOs slide on each other or one RSO is occluded by other objects), our goal is to re-identify the occluded RSO based on the history information. In other words, we aim to identify that a TBD result in the current frame k is the successor of an existing target in the previous frame $k-1$ or a new target that appears in the current frame k. Suppose that we have M_k RFSs in frame k and M_{k-1} singleton RFSs in frame $k-1$. We have also assigned M_{k-1} labels such as $l^1_{k-1}, \ldots, l^{M_{k-1}}_{k-1}$ to each singleton RFS in frame $k-1$ and assigned M_{k-1} scores such as $s^1_{k-1}, \ldots, s^{M_{k-1}}_{k-1}$ to each associated singleton RFS in frame $k-1$. The label is the identification ID for a target, and the score is the total number of times that a target exists in the past $k-1$ frames. We design a labeling management scheme to assign the labels and scores to the M_k RFSs in frame k by assuming that RSOs do not have any significant position changes between two consecutive frames. To this end, we first compute the pairwise Euclidean distance between the average position of all particles in an investigated RFS in frame k with the position of each singleton RFS in frame $k-1$. We then choose the singleton RFS that has the closest distance to the investigated RFS in frame k. If this closest distance is less than a pre-defined threshold (e.g., ten pixels), we consider the chosen singleton RFS in frame $k-1$ and the investigated RFS in frame k as the predecessor and successor pair. Consequently, we assign the investigated RFS the same label as the chosen singleton RFS in the previous frame and assign it a score, whose value is the score value of the chosen singleton RFS plus 1. Otherwise, we assign the investigated RFS a new label and assign its score as 1. This labeling and score assignment is summarized as follows:

$$l^j_k = \begin{cases} l^\theta_{k-1} & \text{if } \underset{1\le i\le M_{k-1}}{min} \{d(Tar^{(i)}_{k-1}, Tar^{(j)}_k)\} \le 10 \\ l^j_k & \text{otherwise} \end{cases} \tag{20}$$

$$s^j_k = \begin{cases} s^\theta_{k-1}+1 & \text{if } \underset{1\le i\le M_{k-1}}{min} \{d(Tar^{(i)}_{k-1}, Tar^{(j)}_k)\} \le 10 \\ 1 & \text{otherwise} \end{cases} \tag{21}$$

where $d(Tar^{(i)}_{k-1}, Tar^{(j)}_k)$ is the Euclidean distance between the average position of all particles in the ith singleton RFS in frame $k-1$ and the average position of all particles in the jth RFS in frame k, and θ is the index of the singleton RFS in frame $k-1$ that has the minimum distance less than ten pixels to the jth RFS in frame k. In other words, $\theta = \arg(\underset{1\le i\le M_{k-1}}{\min} \{d(Tar^{(i)}_{k-1}, Tar^{(j)}_k)\} \le 10)$.

Since CCD imagery data are background cluttered and the RSOs have small SNR, occlusions often happen throughout the frame sequence and significantly decrease the r value of an existing Bernoulli RFS. The decreased r values may lead to the elimination of the existing Bernoulli RFS even when it has been present in the past for a while. The newborn RFSs can eventually detect the occluded RSO right after the occlusion disappears. However, the detected RSO will be considered as a new target and therefore will be assigned a new label, which is undesirable. To address the issues related to the occlusion, we design an occlusion management scheme to employ the history information such as labels and scores obtained in the previous frame to transfer the RFS of an occluded target to a location in the current frame where the target is likely to reappear.

To this end, we check the s value of a target in the previous frame when the r value of the target is significantly decreased and therefore leads to its removal possibly due to occlusion. If this s value is more than a pre-determined threshold (e.g., five repeated times), we keep the RFS of the target in the current frame and transmit this RFS to the next frame by updating the particles and their weights in time. We only repeat this transmission process up to two times if the r value of the target is significantly decreased and its associated s value in the previous frame is bigger than the pre-determined threshold. This setting ensures that the proposed tracking method can effectively label the occluded RSO at the estimated location when the occlusion is abruptly caused by noisy pixels, a common scenario occurred in the CCD imagery data.

It should be noted that the labeled RFS is also used to produce labeled multi-target tracks. Some pertinent examples of labeled RFS include labeled Poisson RFS, labeled multi-Bernoulli (LMB) RFS, and GLMB RFS [20,23,29]. In general, LBM and GLBM filter are more principled approaches than unlabeled multi-Bernoulli filter to track and identify targets in a frame sequence. However, the proposed unlabeled multi-Bernoulli coupled with labeling management has demonstrated a good performance in detecting, tracking, and identifying the RSOs. It assigns accurate labels to the targets that do not have abrupt and drastic changes. Moreover, labeling management is fused with occlusion management, which records history information such as labels and scores, to effectively label the occluded RSOs at the estimated locations when the occlusion is abruptly caused by noisy pixels.

5.3 Noise removal

The aim of this step is to remove the noise captured by a de facto empty RFS, where the likelihood value of each particle representing a noisy pixel is small but their cumulative likelihood value as computed in (18) is big enough to make the existence probability of this empty RFS comparable to the existence probability of a singleton RFS representing a target. In other words, the de facto empty RFS with the high cumulative likelihood value yet small likelihood value for individual particle can be mistakenly considered as a target instead of the noise. To solve this problem, we employ an adaptive likelihood thresholding method to remove such noise by assuming that this kind of noise is chaotic in nature and normally does not repeat itself near its current location in the next frame. To this end, we first compute the likelihood value for each RFS identified in the current frame. We then find all the targets whose s values are larger than 2 (i.e., find all the targets that have been continuously present in the sequence for more than two times) and obtain the minimum likelihood value for these targets. This minimum likelihood value represents the possible dimmest target till the current frame, which normally is higher than the likelihood value of the noise pixel. We finally compute the adaptive likelihood threshold (e.g., T_{adp}) by multiplying this minimum likelihood value by 10^{-3} and remove all RFSs, whose likelihood value is smaller than this adaptive threshold, from the tracking results. This adaptive likelihood thresholding method ensures that all the targets that are repeated more than twice are kept in the tracking results, some potential new targets that are dimmer than our targets will be kept in the tracking results, and the noise pixels that are significantly "dimmer" will be removed from the tracking results.

5.4 Algorithm overview

The algorithm overview of the proposed method for tracking multiple RSOs using multi-Bernoulli filtering is summarized in Algorithm 1.

Algorithm 1 The algorithmic overview

```
Input:
  – A CCD frame sequence
  – $T_{max}$: The maximum number of targets
  – $L_{min}$: The minimum number of particles in a RFS
  – $L_{max}$: The maximum number of particles in a RFS
  – $T_1$: A predefined probability threshold
  – $T_2$: A predefined distance threshold
Output: Tracking results
  1. Define each particle as $[x, y, v_x, v_y]$, where $x$ and $y$ are its location
     and $v_x$ and $v_y$ are its initial velocity at $x$ and $y$ directions.
  2. Set birth RFSs as $\{(r^{(i)}_{\Gamma,k}, p^{(i)}_{\Gamma,k})\}^{4}_{i=1}$ with $r^{(i)}_{\Gamma,k} = 0.25$ for each
     quarter of frame $k$.
  3. For the first frame ($k = 1$)
     (a) $\{\check{\mathbf{x}}^{(i,j)}_{k-1})\}^{L_{max}}_{j=1} \leftarrow$ Randomly generate $L_{max}$ particles for each
         birth RFS
     (b) $\{\mathbf{w}^{(i,j)}_{k-1})\}^{L_{max}}_{j=1} \leftarrow$ Uniformly initialize the weights of the particles.
     (c) Calculate the likelihood values of the particles $g_f(\mathbf{Y}; \check{\mathbf{x}}^{(i,j)}_{k|k-1})$
         using (5).
     (d) $\mathbf{w}^{(i,j)}_{k} \leftarrow$ Update the weights of the particles using (17).
     (e) $\{r^{(i)}_k, p^{(i)}_k\} \leftarrow$ Recalculate the multi-Bernoulli filtering parameters using (18)-(19)
     (f) Resample the particles and restrict the number of particles in
         the range of $[L_{min}, L_{max}]$.
     (g) if $r^{(i)}_k < T_1$
           – Discard $i$th RFS.
     (h) $Tar^{(i)}_k \leftarrow$ Calculate the average position of particles in the
         remaining RFSs as target position.
     (i) if $d(Tar^{(i)}_k, Tar^{(j)}_k) \leq T_2$
           – Merge RFSs $i$ and $j$.
     (j) Keep the first $T_{max}$ RFSs as potential targets.
     (k) $l^i_k \leftarrow$ Assign each target a label using (20).
     (l) $s^i_k \leftarrow$ Assign each target a score using (21).
     (m) Calculate the likelihood value for each target position
         $g_f(\mathbf{Y}; Tar^{(i)}_k)$ using (5).
     (n) Compute the adaptive likelihood threshold $T_{adp}$ as explained
         in Sect. 5.3.
     (o) if $g_f(\mathbf{Y}; Tar^{(i)}_k) \geq T_{adp}$
           – $Tracking\ results \leftarrow Tar^{(i)}_k$
  4. do
     (a) Update the particles of the existing Bernoulli RFSs from the
         previous frame.
     (b) Repeat step 3(a) to produce birth RFSs in current frame.
     (c) Repeat steps 3(b) to 3(o) for the newborn RFSs and the existing
         RFSs.
     until the end of the frame sequence
```

6 Simulation result

We evaluate the proposed multi-Bernoulli RSO tracking method by conducting experiments on various image sequences provided by National Central University Lulin Observatory in Taiwan. These sequences of Earth-orbiting satellites were taken from a ground-based telescope at Lulin Observatory. They were output from a one-megapixel CCD sensor at 16-bit intensity depth. Since the satellites were rate-tracked based on a priori information, they appear as point detections, whereas celestial objects in the background appear as streaks. The sensor consists of 50 cm aperture telescope and $2k \times 2k$ cooled CCD with the field of view of $1.74° \times 1.78°$, which is equivalent to 3 arcsec/pixel resolution. More information about our data set can be found in [26]. All computations are performed in MATLAB using Intel®-Core$^{\text{TM}}$ i7-3370 (3.4 GHz) system with 16 GB RAM.

We conduct the experiments on ten image sequences, wherein the experts provide the ground-truth information

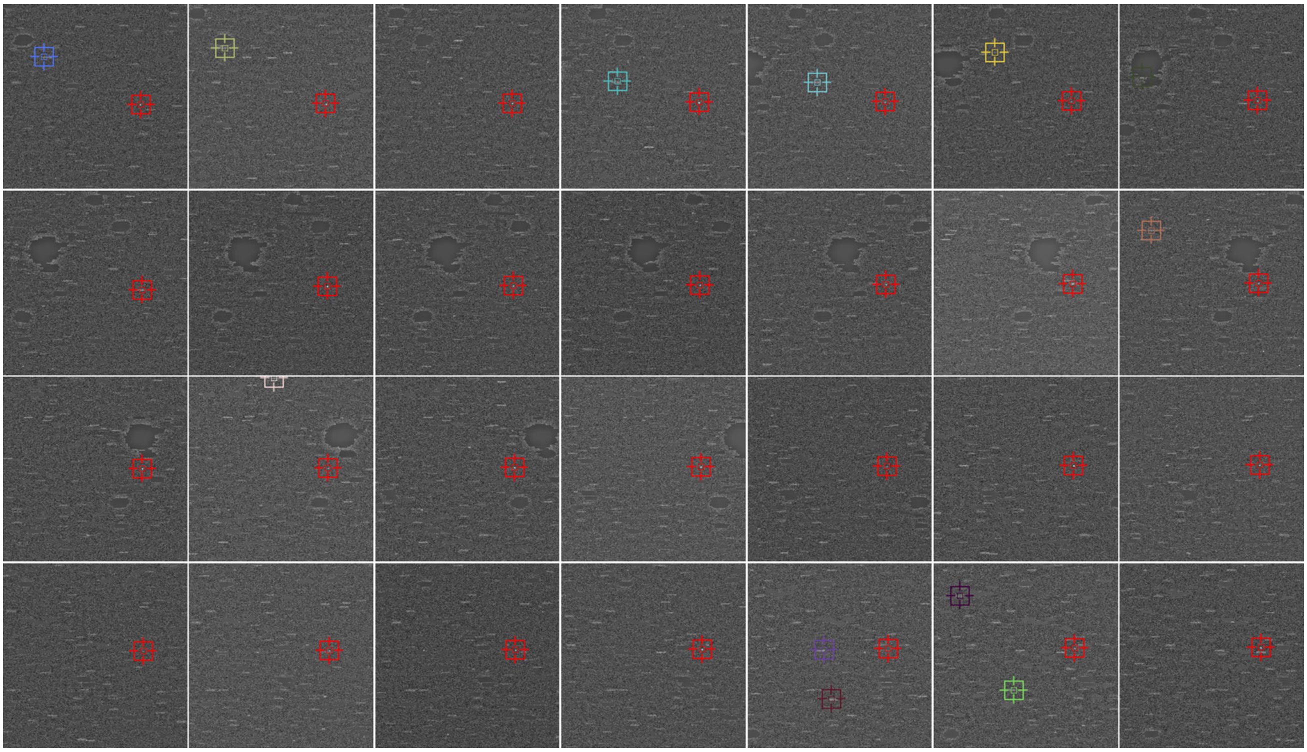


Fig. 3 TBD results for deb020, where each frame is sequentially shown in the raster scan order. The mean of each RFS is highlighted with crosshair. The red rectangle shows the tracking results for the RSO. The non-red rectangles represent the false objects that the algorithm detects (color figure online)

for each RSO. Three of these ten sequences (e.g., deb020, deb021, and deb024) contain exactly one RSO. The remaining seven image sequences (e.g., deb059, deb032, deb029, deb030, deb033, deb043, and deb045) contain multiple RSOs, which might enter into, leave from, or move around at any frame of the sequences. In this section, we provide the detailed information and both quantitative and qualitative analyses for three image sequences titled deb020, deb059, and deb032. In addition, we provide the quantitative results for other image sequences in terms of the number of particles, the average running time, the average estimation error, the average OSPA distance, the average localization error, and the average cardinality error.

We present the experimental results for "deb020" to evaluate the performance of the proposed method to detect a single RSO without knowing this explicit information. The frames in this sequence are cropped from their original data to contain exactly one RSO. The size of each cropped frame is 301×301, and there are 28 images in the sequence. In this cropped image sequence, the dim RSO is of the average SNR value of 9.0095 dB and appears in the first frame and occluded by the noise in the 26th frame. We initialize $L_{\max}$ =2500 number of particles for each Birth Bernoulli RFS, and the number of particles is constrained between $L_{\min}$ =1000 and $L_{\max}$ =2500 later on to avoid high computational cost, as summarized in Algorithm 1. The particles are resampled in each iteration when processing this sequence. The implemented method processes two frames per minute. Figure 3 presents the tracking results for each frame in this cropped image sequence, where the average position of particles in each Bernoulli RFS that might contain a target (i.e., its RFS component with a probability of existence above the adaptive likelihood threshold computed in Sect. 5.3) is plotted in different colors on top of each frame. It can be easily inferred that the Bernoulli RFS containing the RSO is marked in red since it is present in all frames with the expected trajectory. The proposed tracking method successfully handles the occlusion occurred in frame 26 and continues tracking the same object as marked in red in the remaining frames. The tracking method also detects some objects as potential targets due to its similarity to the RSO or the relative high likelihood value totaled from the small likelihood values of all particles within a RFS. The mean position values of all the particles in the associated RFSs for these potential targets are shown as crosshairs of the rectangles in different colors. Since they are not RSOs, the tracking method does not track these counterfeit objects.

We present the experimental results for "deb059" to evaluate the performance of the proposed method to detect two RSOs without knowing this explicit information. This sequence contains 28 frames with the size of 1024 × 1024 pixels for each frame. The two RSOs appear in this sequence

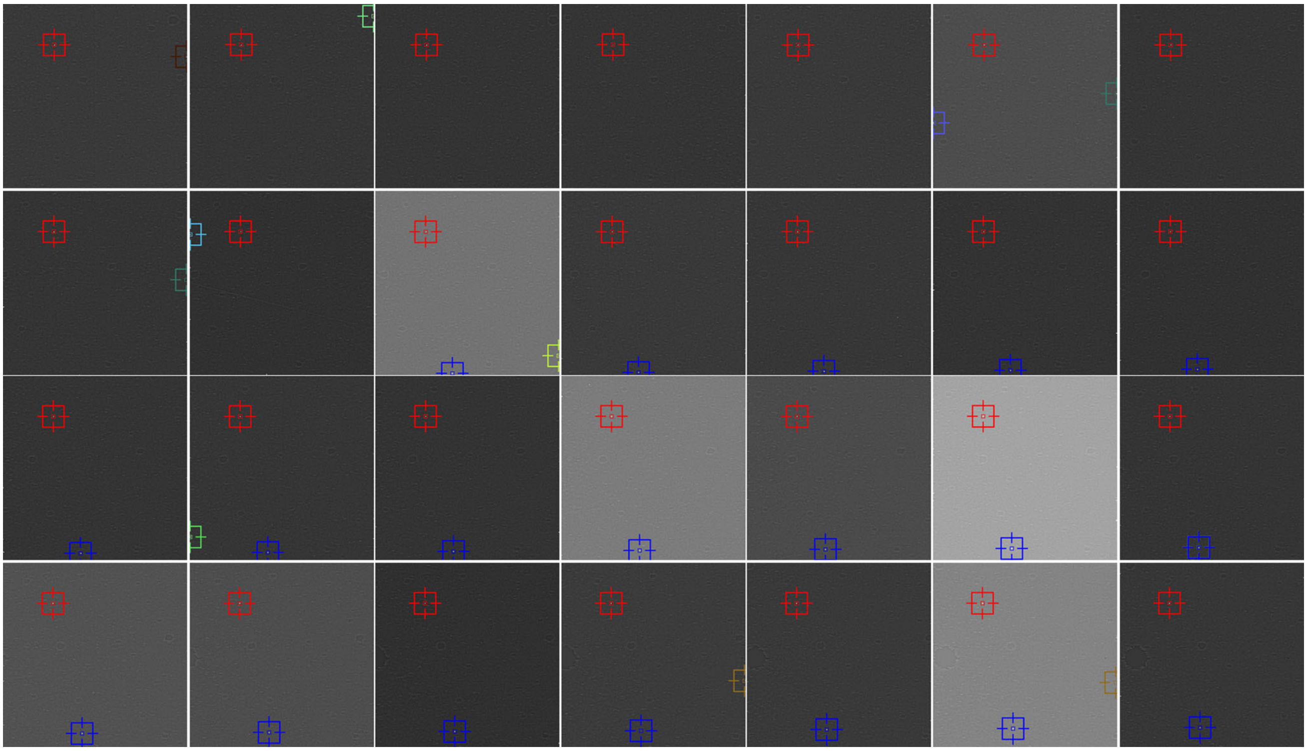


Fig. 4 TBD results for deb059, where each frame is sequentially shown in the raster scan order. The mean of each RFS is highlighted with crosshair. The red rectangle shows the tracking results for the first RSO. The blue rectangle shows the tracking results for the second RSO (color figure online)

with the average SNR value of 18.023 dB and 19.027 dB, respectively. The first RSO appears in all frames with partial occlusion at the fourth frame, and the second RSO completely comes into view in the tenth frame and survives till the end of sequence with occlusion occurred at the 25th frame. Each RSO has different average velocities throughout the time, and the second RSO has higher velocities at both directions. Since the size of each frame in this data set is approximately 11 times larger than the size of each frame in the first data set, we initiated $L_{\max}$ =25,000 number of particles for each of the birth Bernoulli RFSs and constrained the number of particles between $L_{\min}$ =1000 and $L_{\max}$ =25,000 to ensure enough coverage of the potential targets. Similarly, the particles are resampled in each iteration when processing this sequence. The implemented method processes 0.5 frame per minute for this range of number of particles. Figure 4 presents the tracking results for each frame in "deb059" sequence, where the average position of particles in each Bernoulli RFS that might contain a target is plotted in different colors on top of each frame. It can be easily inferred that the Bernoulli RFS containing the first RSO is marked in red since it is present in all frames with slower movement. The Bernoulli RFS containing the second RSO is marked in blue since it shows up in the tenth frame based on the birth RFS and is consistently present in the remaining frames based on the existing RFS. The proposed method successfully handles the occlusion occurred at the different frames and continues tracking the same RSOs with their associated labels as marked in red and blue in the remaining frames. Similar to the first track, the tracking method also detects some objects as potential RSOs as shown in frames 1, 2, 6, 8, 9, 10, 16, 25, and 27. Since they are not RSOs, the tracking method does not keep track of them. In addition, these non-RSO objects do not repeatedly appear in the sequence since they are represented in different colors. As a result, we can safely say that the proposed tracking system succeeds in simultaneously detecting and tracking the two RSOs, which have different velocities in x and y directions, even when there are drastic changes in background and target intensity across images.

We present the experimental results for "deb032" image sequence to evaluate the performance of the proposed method to detect two RSOs without knowing this explicit information. This sequence contains 28 frames with the size of 1024×1024 pixels for each frame. The two RSOs appear in this sequence with the average SNR value of 18.042 and 19.016 dB, respectively. The first RSO appears in the first frame and exists throughout the image sequence with occasional occlusion at frames 5, 16, and 24. The second RSO completely appears in the ninth frame and exists till the end of the sequence. The two RSOs have different velocities compared to each other. Their velocities are also different from the velocities of the RSOs in the previous two tracks. We

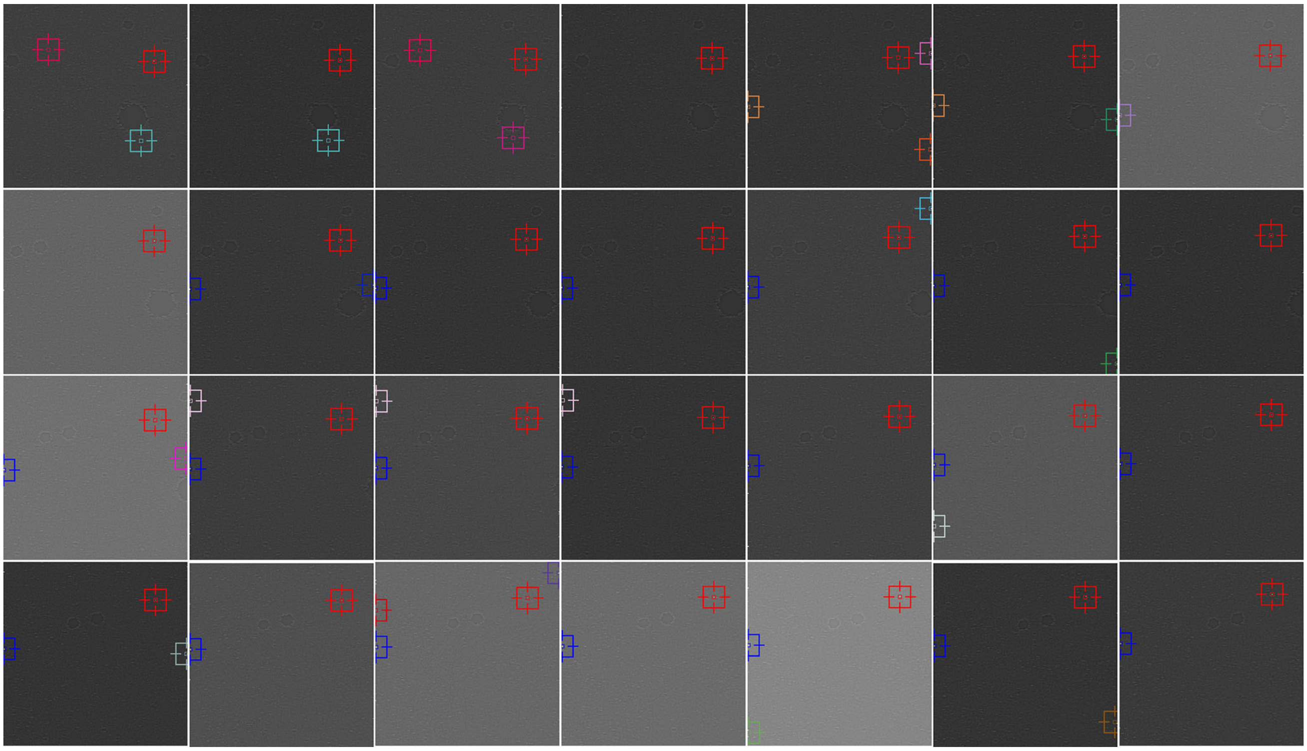


Fig. 5 TBD results for deb032, where each frame is sequentially shown in the raster scan order. The mean of each RFS is highlighted with crosshair. The red rectangle shows the tracking results for the first RSO. The blue rectangle shows the tracking results for the second RSO (color figure online)

use the same setting in the second experiment to constrain the number of particles for each of the Bernoulli RFSs and resample the particles in each iteration. Figure 5 presents the tracking results for each frame in "deb032" sequence, where the average position of particles in each Bernoulli RFS that might contain a target is plotted in different colors on top of each frame. It can be easily inferred that the Bernoulli RFS containing the first RSO is marked in red since it is present in all frames. The Bernoulli RFS containing the second RSO is marked in blue since it shows up in the ninth frame based on the birth RFS and is consistently present in the remaining frames based on the existing RFS. Although the first RSO is occluded in some frames and the second RSO moves along the border of the images, the proposed tracking method successfully handles these situations and continues tracking the same RSOs with their associated labels as marked in red and blue in the remaining frames. As expected, the tracking method also detects some objects as potential RSOs and does not keep track of them. Therefore, we can safely say that the proposed tracking system succeeds in simultaneously detecting and tracking the two RSOs, which have different velocities at x and y directions.

We use the distance between the locations of inlier estimated RSOs and the locations of their corresponding ground-truth RSOs to compute the location estimation error since the localization error is important in the SDA applications. Figure 6 shows the accuracy of the tracking results for each frame of the deb020, deb059, and deb032 sequences in terms of the location estimation error in pixels. It clearly shows that the tracking location estimation error for the deb020 sequence is much smaller and less fluctuated compared to the tracking location estimation error for the other two sequences, which contain two RSOs. For example, 25 frames in the deb020 sequence have the location estimation error less than one pixel, two frames have the location estimation error between one and two pixels, and one frame has the location estimation error around five pixels. However, for the other two sequences, most frames have the location estimation error less than two pixels with approximately a half of these frames having the location estimation error less than one pixel.

Figure 7 shows the accuracy of the tracking results for each frame of the deb020, deb059, and deb032 sequences in terms of the estimation error between the velocity of the inlier estimated RSOs at both x and y directions and the actual velocity of their corresponding ground-truth RSOs. It clearly shows that the tracking velocity estimation error for the deb020 sequence is much smaller and less fluctuated compared to the tracking velocity estimation error for the other two sequences. All but one frame in the deb020 sequence have the velocity estimation error of less than 0.5 pixel per frame at the x direction and less than one pixel per frame at

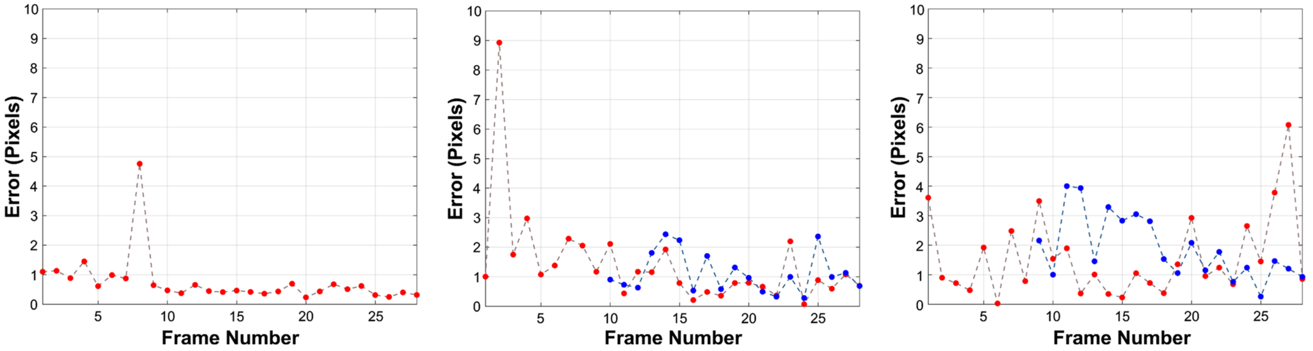


Fig. 6 Tracking estimation error in RSO location for each frame in three image sequences. First column: one RSO in deb020; second column: two RSOs in deb059 (RSO1 in red and RSO2 in blue); third column: two RSOs in deb032 (RSO1 in red and RSO2 in blue) (color figure online)

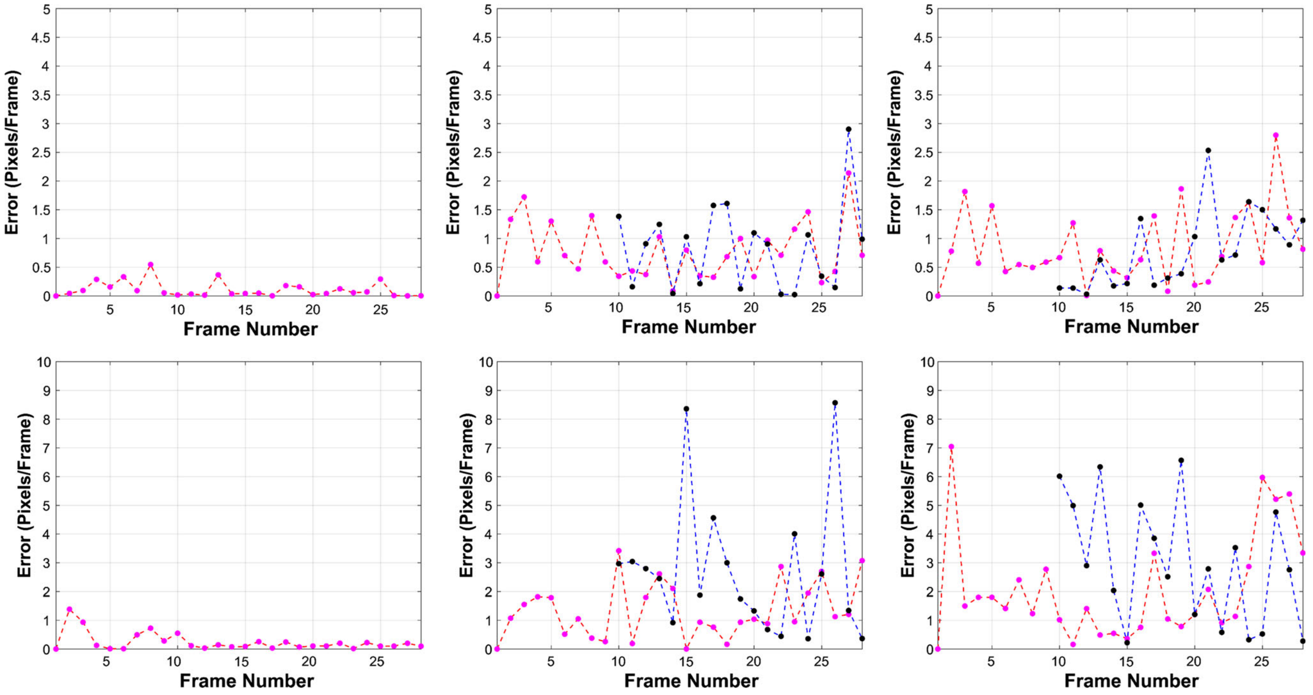


Fig. 7 Tracking estimation error in RSO velocity for each frame in three image sequences at *x* direction (top row) and *y* direction (bottom row). First column: one RSO in deb020; second column: two RSOs in deb059 (RSO1 in red and RSO2 in blue); third column: two RSOs in deb032 (RSO1 in red and RSO2 in blue) (color figure online)

the *y* direction. Most frames in the other two sequences have the velocity estimation error of more than 0.5 pixel per frame at the *x* direction and more than one pixel per frame at the *y* direction.

To further evaluate the proposed method, we provide the quantitative results for all image sequences in Table 1. This table summarizes the maximum number of particles for each Bernoulli RFS, the average running time, the average estimation error in pixels between the locations of inlier estimated RSOs and the locations of their corresponding ground-truth RSOs, and the average estimation error in pixels per frame between the velocity of inlier estimated RSOs at both *x* and *y* directions and the velocity of their corresponding ground-truth RSOs for all experiments. Similar to the cropping of deb020, the frames in the sequences deb021 and deb024 are intentionally cropped from their original data to contain exactly one RSO. The size of each cropped frame is 301 × 301, and the number of frames in the sequence is 28. The other seven sequences contain 28 frames with the size of 1024 × 1024 pixels for each frame. As shown in Table 1, the average running time for processing the deb020, deb021, and deb024 sequences is four times faster than the average running time for processing the other seven sequences mainly due to the significantly smaller number of particles per RFS. The average estimation error for the single RSO location in the three sequences (deb020, deb021, and deb024) is less

Table 1 Summary of the ten experiments in terms of the particle numbers, the average running time, and the average estimation errors

Image sequence name	# of Particles Per RFS	Average running time (Frames/s)	RSOs	Average estimation errors		
				Location (Pixels)	x-velocity (Pixels/Frame)	y-velocity (Pixels/Frame)
deb020	2500	2	One RSO	0.7642	0.1116	0.2837
deb021	2500	2	One RSO	0.8431	0.2512	0.3315
deb024	2500	2	One RSO	0.9545	0.2121	0.4549
Average	2500	2		0.8539	0.1916	0.3567
deb059	25,000	0.5	RSO1	1.4052	0.7742	1.3248
			RSO2	1.1096	0.8305	2.7045
deb032	25,000	0.5	RSO1	1.5661	0.8535	2.0686
			RSO2	1.3549	0.7876	3.0080
deb029	25,000	0.5	RSO1	1.5613	0.8112	1.9156
			RSO2	1.1567	0.7876	3.0080
deb030	25,000	0.5	RSO1	1.1286	0.5154	1.7815
			RSO2	0.9859	0.6371	2.0963
deb033	25,000	0.5	RSO1	1.0593	0.9845	1.8215
			RSO2	1.1575	1.1852	2.5989
Average	25,000	0.5		1.2485	0.8167	2.2328
deb043	25,000	0.5	RSO1	0.8565	0.7893	1.5496
			RSO2	1.2678	0.8979	2.1315
			RSO3	1.5657	0.9613	2.7545
deb045	25,000	0.5	RSO1	1.3215	0.9652	1.3315
			RSO2	1.5416	0.9313	2.0215
			RSO3	0.7634	0.6895	1.1543
Average	25,000	0.5		1.2194	0.8724	1.8238

than one pixel while the average estimation error for the locations of both RSO1 and RSO2 in the five sequences (deb059, deb032, deb029, deb030, and deb033) is less than two pixels. The average estimation error for the locations of RSO1, RSO2, and RSO3 in the two sequences (deb043 and deb045) is less than two pixels. The average estimation errors in terms of x velocities and y velocities for all RSOs in the ten sequences are less than one pixel per frame and two pixels per frame, respectively. The small location and velocity estimation errors indicate the efficiency of the proposed tracker to accurately detect and track multiple RSOs while not knowing any explicit a priori information.

We further use the OSPA distance [25] between the states of the estimated RSOs (including both inliers and outliers) and the states of the ground-truth RSOs in each of the ten frame sequences to provide supplementary evaluation of the proposed multi-RSOs tracking framework. The OSPA distance is a meaningful mathematical metric to jointly calculate differences in cardinality and localization between the states of individual targets in two finite sets. For two finite sets $X = \{x_1, \ldots, x_{n_x}\} \subset \mathcal{X}$ and $Z = \{z_1, \ldots, z_{n_z}\} \subset \mathcal{Z}$, a distance between a target $x_k \in X$ and a target $z_{k'} \in Z$ is defined as $d^{(c)}(x_k, z_{k'}) = \min(c, \|x_k - z_{k'}\|)$, where c is an empirically determined cutoff positive value and $\|\cdot\|$ represents the Euclidean distance. The OSPA distance metric [25] for the finite sets X and Z is computed as follows:

$$\bar{d}_p^{(c)}(X, Z) = \Big(\big(\frac{1}{n_z}\big(\underset{\pi\in\Pi_{n_z}}{min} \sum_{i=1}^{n_x} d^{(c)}(x_i, z_{\pi(i)})^p + c^p(n_z - n_x)\big)\Big)^{\frac{1}{p}} \tag{22}$$

when $n_x \leq n_z$. If $n_x > n_z$, $\bar{d}_p^{(c)}(X, Z) = \bar{d}_p^{(c)}(Z, X)$. If $n_x = n_z = 0$, $\bar{d}_p^{(c)}(X, Z) = \bar{d}_p^{(c)}(Z, X) = 0$. Here, p is a positive value greater than or equal to 1 and determines the sensitivity of $\bar{d}_p^{(c)}$ to outlier estimates, and Π_{n_z} is the set of permutations on $\{1, 2, \ldots, n_z\}$ for any positive integer n_z. For the multi-target filtering cases, the OSPA metric [25] can be interpreted as a pth order "per-object" error, consisting of two components, which separately justify the "localization" and "cardinality", as follows:

$$\bar{e}_{p,\mathrm{loc}}^{(c)}(X, Z) = \Big(\frac{1}{n_z}\big(\min_{\pi\in\Pi_{n_z}} \sum_{i=1}^{n_x} d^{(c)}(x_i, z_{\pi(i)})^p\Big)^{\frac{1}{p}} \tag{23}$$

$$\bar{e}_{p,\mathrm{card}}^{(c)}(X, Z) = \Big(\frac{1}{n_z}\big(c^p(n_z - n_x)\Big)^{\frac{1}{p}} \tag{24}$$

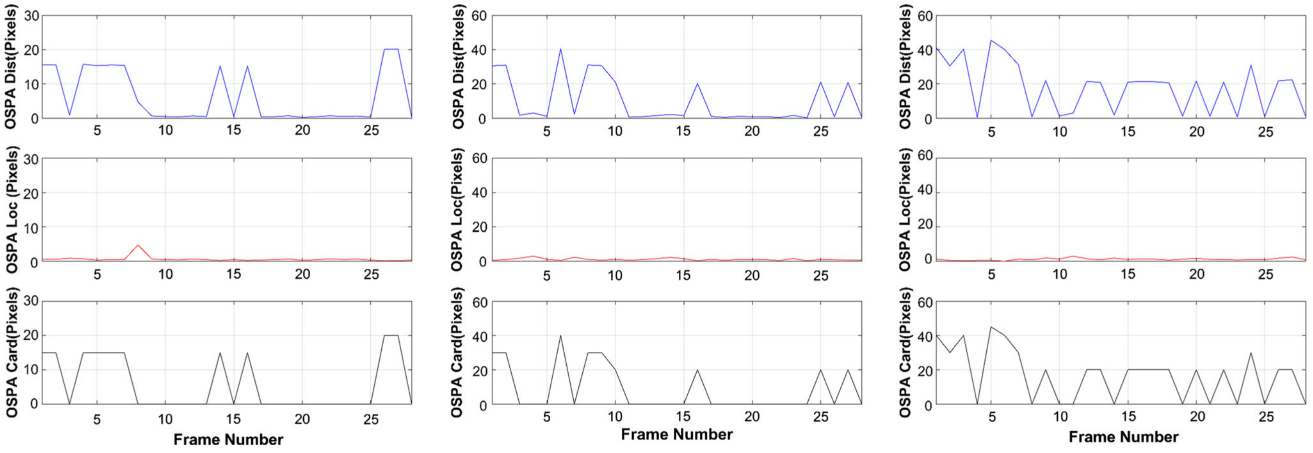


Fig. 8 OSPA distance together with its localization and cardinality components. First column: deb020 ($c = 30$, $p = 1$); second column: deb059 ($c = 60$, $p = 1$); third column: deb032 ($c = 60$, $p = 1$)

Table 2 Summary of ten experiments in terms of the average OSPA distance, the average localization error, and the average cardinality error (all in pixels)

Image sequence name	Average OSPA distance	Average localization error	Average cardinality error
deb020	6.32	0.60	5.71
deb021	7.15	0.72	6.43
deb024	8.78	0.85	7.93
deb059	8.18	1.04	7.14
deb032	15.27	1.13	14.13
deb029	9.76	0.92	8.84
deb030	10.52	1.01	9.51
deb033	8.27	1.01	7.26
deb043	12.59	1.08	11.51
deb045	14.19	1.21	12.98
Average	10.10	0.95	9.14

when $n_x \leq n_z$. If $n_x > n_z$, $\bar{e}^{(c)}_{p,\mathrm{loc}}(X, Z) = \bar{e}^{(c)}_{p,\mathrm{loc}}(Z, X)$ and $\bar{e}^{(c)}_{p,\mathrm{card}}(X, Z) = \bar{e}^{(c)}_{p,\mathrm{card}}(Z, X)$.

In our calculation, we choose the order of p as 1 to facilitate a direct interpretation of the OSPA metric and its localization and cardinality components. The value of cut-off c determines the relative weights of the components of the localization error and the cardinality error as a part of the total error. The smaller value of c mostly emphasizes the localization error and makes the OSPA metric less sensitive to the cardinality error. On the other hand, the larger value of c mostly emphasizes the cardinality error and makes the OSPA metric less sensitive to the localization error. We employ different c values for the image sequences based on the size of the images. Specifically, we use $c = 30$ pixels for the deb020, deb021, and deb024 image sequences and use $c = 60$ pixels for the other seven image sequences as moderate choices to maintain a balance between the localization and cardinality errors.

For each image in the ten frame sequences, we calculate the OSPA metric (i.e., $\bar{d}^{(c)}_p$, $\bar{e}^{(c)}_{p,\mathrm{loc}}$, and $\bar{e}^{(c)}_{p,\mathrm{card}}$) between the estimated locations of objects and the locations of ground-truth RSOs. In Fig. 8, we only provide the diagram of OSPA distance (i.e., $\bar{d}^{(c)}_p$) (in blue), the localization error (i.e., $\bar{e}^{(c)}_{p,\mathrm{loc}}$) (in red), and the cardinality error (i.e., $\bar{e}^{(c)}_{p,\mathrm{card}}$) (in black) per frame for the three sequences (i.e., deb020, deb059, and deb032). The average OSPA distance is 6.32 pixels for deb020, 8.18 pixels for deb059, and 15.27 pixels for deb032. The average localization error is 0.60 pixels for deb020, 1.04 pixels for deb059, and 1.13 pixels for deb032. The average cardinality error is 5.71 pixels for deb020, 7.14 pixels for deb059, and 14.13 pixels for deb032.

Table 2 summarizes the average OSPA distance $\bar{d}^{(c)}_p$, average localization error $\bar{e}^{(c)}_{p,\mathrm{loc}}$, and average cardinality error $\bar{e}^{(c)}_{p,\mathrm{card}}$ for the ten image sequences. The average OSPA distance, the average localization error, and the average cardinality error for all RSOs in the ten image sequences are 10.10 pixels, 0.95 pixels, and 9.14 pixels, respectively. It clearly shows that the proposed tracking method has significantly

Table 3 Comparison of the proposed method with P-TBD and Bernoulli-P-TBD methods in terms of the average location, the average x-velocity, and the average y-velocity

Methods name	Average estimation error								
	Location (Pixels)			x-velocity (Pixels/Frame)			y-velocity (Pixels/Frame)		
	deb020	deb021	deb024	deb020	deb021	deb024	deb020	deb021	deb024
P-TBD [26]	1.2415	1.1582	1.196	0.3014	0.2632	0.4001	0.4018	0.4534	0.5191
Bernoulli-P-TBD [5]	1.015	1.0961	1.012	0.2082	**0.2207**	0.3001	0.3173	**0.3012**	0.5093
Proposed method	**0.7642**	**0.8431**	**0.9545**	**0.1116**	0.2512	**0.2121**	**0.2837**	0.3315	**0.4549**

small average localization error, which indicates the robustness of the method in localizing the RSOs. The proposed tracking method yields relatively larger OSPA distance and large cardinality errors in some frames mainly due to detection of counterfeit point objects similar to RSOs. However, the proposed algorithm does not keep track of these fake targets throughout the sequences. It should be noted that the cardinality error is 0 in most frames, which means that all the RSOs are correctly identified and no more counterfeit objects are identified. As a result, the OSPA distance is small in these frames. In summary, we conclude that the proposed tracking method achieves accurate localization performance in all frames and accurate cardinality performance in a majority of frames.

To the best of our knowledge, this is the first attempt to apply the multi-Bernoulli RFS framework in SDA to track multiple RSOs in clutters in telescope imagery data with low SNR. State-of-the-art RSO tracking algorithms [5,26] are only applicable for detecting and tracking a single RSO instead of multiple RSOs in frame sequences. Roughly speaking, the proposed multi-RSOs tracking method (i.e., multi-Bernoulli-TBD) is a robust extension of the single RSO tracking methods (i.e., P-TBD [26] and Bernoulli-P-TBD [5]). It can be reliably applied to track both single RSO and multiple RSOs without knowing any prior information about the number of RSOs and their initial locations in the sequences. Uetsuhara and Ikoma [26] and Fujimoto et al. [5] use the same settings as our experiments to evaluate the performance of their proposed methods to track the single RSO in three datasets (e.g., deb020, deb021, and deb024). Specifically, the P-TBD method needs the initial location of the single RSO in all three image sequences to track the single RSO. Similarly, the Bernoulli-P-TBD method needs the initial location of the single RSO in deb021 and deb024 sequences to track the RSO while tracking the RSO in the deb020 sequence without knowing its initial location. The proposed method does not need the initial location of each RSO as prior information since it uses the multi-Bernoulli filter with birth RFSs to automatically detect the target as it appears in any location of each frame. We simulate the P-TBD and Bernoulli-P-TBD methods in MATLAB and run them on the three datasets (e.g., deb020, deb021, and deb024) using the same parameters described in their respective research papers. Table 3 compares the performance of the proposed method, the P-TBD method, and the Bernoulli-P-TBD method in terms of the average location estimation error in pixels and the average estimation error of velocity in pixels along x and y directions per frame. We show the smallest estimation error for each dataset in bold. It clearly shows that the proposed method achieves the best performance in terms of the average location estimation error for all three sequences. It also yields the best performance in terms of the average x-velocity estimation error and the average y-velocity estimation error for deb020 and deb024 sequences. For deb021 sequence, the proposed method achieves the second best performance in terms of the average x-velocity estimation error and the average y-velocity estimation error. The best performance is achieved by the Bernoulli-P-TBD method with the known initial position of the single RSO. It should be mentioned that the suggested number of particles is 10^4 in P-TBD, which approximately processes one frame per second for each of the three frame sequences. The Bernoulli-P-TBD processes two frames per second with 10^4 particles. In this method, the suggested number of particles is different for each of CCD frame sequences. For deb020, deb021, and deb024, the number of particles is 10^4, 2×10^4, and 2×10^4, respectively. Consequently, the method processes two frames per second for deb020 sequence and one frame per second for deb021 and deb024 sequences. Our proposed method does not require any prior information about the initial locations of the RSOs in the sequences and processes two frames per second for each of the deb020, deb021, and deb024 frame sequences, which is faster than the two other compared methods.

All the experimental results clearly demonstrate that the proposed multi-Bernoulli-TBD method, without knowing any prior information about the number of RSOs, is able to successfully track multiple RSOs in clutters in imagery data with low SNR. When tracking a single RSO, the proposed method, without knowing any prior information about the initial location of the RSO in the sequences, achieves better performance than the two state-of-the-art RSO tracking methods, which are applicable for detecting and tracking a single RSO by using the initial location of the RSO in the first frame.

7 Conclusions

In this paper, we propose a fast and reliable TBD approach to simultaneously detect, track, and identify an unknown and variable number of RSOs without any prior information and any explicit detection. We evaluate the proposed tracking method on ten frame sequences provided by National Central University Lulin Observatory in Taiwan. The experimental results clearly show that the proposed tracking system succeeds in simultaneously detecting and tracking multiple RSOs with different velocities at x and y directions. To quantitatively evaluate the performance of the proposed multi-RSO tracker, we compute the location estimation error in pixels and the OSPA distance together with the OSPA localization error and the OSPA cardinality error in pixels for each frame in the frame sequences. The extensive experimental results demonstrate that the proposed multi-RSO tracking method achieves accurate localization performance in all frames and accurate cardinality performance in a majority of frames. In summary, our contributions are:

- Introducing a PSF-based separable likelihood function that is capable of making an articulate distinction between RSOs and any spurious signals in background cluttered CCD imagery data.
- Implementing the RFS-based multi-Bernoulli filtering framework by using the SMC method to refine and reject the measurement likelihood function for simultaneously detecting and localizing multiple RSOs without any prior information.
- Incorporating the likelihood function in the observation model to update the multi-Bernoulli parameters forward in time and decide which RFS contains a target based on the updated parameters.
- Designing a labeling and occlusion management strategy to find the predecessor and successor pairs between the previous and current frames and transmit the RFS of an occluded target forward to the next frame to address the occlusion issues.
- Employing an adaptive thresholding method, which utilizes the history information of the presence or absence of a target to remove the chaotic noise.

We will further improve the likelihood function by using computer vision techniques to detect and track RSOs with smaller SNR values. In addition, we will parallelize the filter to reduce the computational time to handle an immense number of targets. We will also investigate the employment of machine learning techniques or an appropriate adaptive threshold in the merging process to relax false positive merging events. Finally, we will consider implementing the multi-Bernoulli filtering using the Gaussian mixture model instead of the SMC method to reduce the computational time.

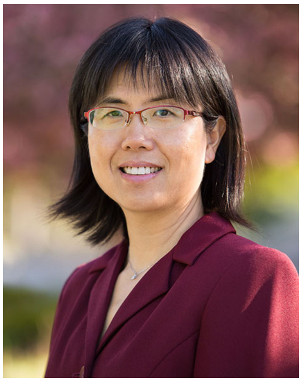
Acknowledgements The authors would like to thank Dr. Kohei Fujimoto for providing the dataset and his constructive suggestions during the planning and development of this research work.

References

1. Buzzi, S., Lops, M., Venturino, L., Ferri, M.: Track-before-detect procedures in a multi-target environment. IEEE Trans. Aerosp. Electron. Syst. **44**(3), 1135–1150 (2008)
2. Cox, I.J., Hingorani, S.L.: An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. IEEE Trans. Pattern Anal. Mach. Intell. **18**(2), 138–150 (1996)
3. Czyz, J., Ristic, B., Macq, B.: A particle filter for joint detection and tracking of color objects. Image Vis. Comput. **25**(8), 1271–1281 (2007)
4. Flewelling, B., Sease, B.: Computer vision techniques applied to space object detect, track, id, and characterize. Technical report, Air Force Research Lab Kirtland AFB NM (2014)
5. Fujimoto, K., Uetsuhara, M., Yanagisawa, T.: Statistical track-before-detect methods applied to faint optical observations of resident space objects. In: Proceedings of the Advanced Maui Optical and Space Surveillance Technical Conference (2015)
6. Goodman, I.R., Mahler, R.P., Nguyen, H.T.: Mathematics of Data Fusion, vol. 37. Springer, Berlin (2013)
7. Hoseinnezhad, R., Vo, B.N., Vo, B.T.: Visual tracking in background subtracted image sequences via multi-bernoulli filtering. IEEE Trans. Signal Process. **61**(2), 392–397 (2013)
8. Hoseinnezhad, R., Vo, B.N., Vo, B.T., Suter, D.: Visual tracking of numerous targets via multi-Bernoulli filtering of image data. Pattern Recognit. **45**(10), 3625–3635 (2012)
9. Isard, M., MacCormick, J.: Bramble: a Bayesian multiple-blob tracker. In: Proceedings of the International Conference on Computer Vision (ICCV), vol. 2, pp. 34–41. IEEE (2001)
10. Karttunen, H., Kröger, P., Oja, H., Poutanen, M., Donner, K.J.: Fundamental Astronomy. Springer, Berlin (2016)
11. Koblick, D., Goldsmith, A., Klug, M., Mangus, P., Flewelling, B., Jah, M., Shanks, J., Piña, R., Stauch, J., Baldwin, J.: Ground optical signal processing architecture for contributing space-based SSA sensor data. Technical report, Air Force Research Lab Kirtland AFB NM (2014)
12. Kristan, M., Perš, J., Perše, M., Kovačič, S.: Closed-world tracking of multiple interacting targets for indoor-sports applications. Comput. Vis. Image Underst. **113**(5), 598–611 (2009)
13. Mahler, R.: Phd filters of higher order in target number. IEEE Trans. Aerosp. Electr. Syst. **43**(4), 1523–1543 (2007)
14. Mahler, R.P.: Multitarget bayes filtering via first-order multitarget moments. IEEE Trans. Aerosp. Electron. Syst. **39**(4), 1152–1178 (2003)
15. Mahler, R.P.: Statistical Multisource–Multitarget Information Fusion. Artech House, Inc, Norwood (2007)
16. Mahler, R.P.: Advances in Statistical Multisource–Multitarget Information Fusion. Artech House, Norwood (2014)
17. Murphy, T.S., Holzinger, M.J., Flewelling, B.: Orbit determination for partially understood object via matched filter bank. In: AAS/AIAA Astrodynamics Specialists Meeting (2015)

18. Nummiaro, K., Koller-Meier, E., Van Gool, L.: Object tracking with an adaptive color-based particle filter. Pattern Recognit. **2449**, 353–360 (2002)
19. Okuma, K., Taleghani, A., De Freitas, N., Little, J.J., Lowe, D.G.: A boosted particle filter: multitarget detection and tracking. In: European Conference on Computer Vision, pp. 28–39. Springer (2004)
20. Papi, F., Vo, B.N., Vo, B.T., Fantacci, C., Beard, M.: Generalized labeled multi-Bernoulli approximation of multi-object densities. IEEE Trans. Signal Process. **63**(20), 5487–5497 (2015)
21. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: European Conference on Computer Vision, pp. 661–675. Springer (2002)
22. Reid, D.: An algorithm for tracking multiple targets. IEEE Trans. Autom. Control **24**(6), 843–854 (1979)
23. Reuter, S., Vo, B.T., Vo, B.N., Dietmayer, K.: The labeled multi-Bernoulli filter. IEEE Trans. Signal Process. **62**(12), 3246–3260 (2014)
24. Schildknecht, T., Ploner, M., Hugentobler, U.: The search for debris in GEO. Adv. Space Res. **28**(9), 1291–1299 (2001)
25. Schuhmacher, D., Vo, B.T., Vo, B.N.: A consistent metric for performance evaluation of multi-object filters. IEEE Trans. Signal Process. **56**(8), 3447–3457 (2008)
26. Uetsuhara, M., Ikoma, N.: Faint debris detection by particle based track-before-detect method. In: Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference (2014)
27. Vermaak, J., Maskell, S., Briers, M., Pérez, P.: Bayesian visual tracking with existence process. In: Proceedings of the International Conference on Image Processing (ICIP), vol. 1, pp. I–721. IEEE (2005)
28. Vo, B.N., Vo, B.T., Pham, N.T., Suter, D.: Joint detection and estimation of multiple objects from image observations. IEEE Trans. Signal Process. **58**(10), 5129–5141 (2010)
29. Vo, B.T., Vo, B.N.: Labeled random finite sets and multi-object conjugate priors. IEEE Trans. Signal Process. **61**(13), 3460–3475 (2013)
30. Vo, B.T., Vo, B.N., Cantoni, A.: Analytic implementations of the cardinalized probability hypothesis density filter. IEEE Trans. Signal Process. **55**(7), 3553–3567 (2007)
31. Vo, B.T., Vo, B.N., Cantoni, A.: The cardinality balanced multi-target multi-Bernoulli filter and its implementations. IEEE Trans. Signal Process. **57**(2), 409–423 (2009)
32. Vo, B.T., Vo, B.N., Hoseinnezhad, R., Mahler, R.P.: Robust multi-Bernoulli filtering. IEEE J. Sel. Top. Signal Process. **7**(3), 399–409 (2013)
33. Yanagisawa, T., Kurosaki, H., Oda, H., Tagawa, M.: Ground-based optical observation system for LEO objects. Adv. Space Res. **56**(3), 414–420 (2015)
34. Yanagisawa, T., Nakajima, A., Kimura, T.: The stacking method for detection of small GEO debris and moving objects. Proc. Int. Symp. Space Technol. Sci. **23**, 2324–2329 (2002)
35. Yang, M., Yu, T., Wu, Y.: Game-theoretic multiple target tracking. In: Proceedings of the International Conference on Computer Vision (ICCV), pp. 1–8. IEEE (2007)

Mohammadreza Javanmardi received the B.S. degree in Electrical Engineering from Shiraz University in 2011 and the M.S. degree in Electrical Engineering from Shiraz University in 2014. He is currently a Ph.D. student in the Computer Science Department at Utah State University. He works as a research assistant at Computer Vision Laboratory of Utah State University. His research interests are in image processing, optimization, deep learning, estimation, and 3-D point cloud processing.



Xiaojun Qi received the B.S. degree in Computer Science from Donghua University in 1993, the M.S. degree in Computer Science from Shenyang Institute of Automation, the Chinese Academy of Sciences in 1996, and the Ph.D. degree in Computer Science from Louisiana State University in 2001. In 2002, she joined the Department of Computer Science at Utah State University as a tenure-track Assistant Professor. In 2008, she received tenure and was promoted to Associate Professor. In 2015, she was promoted to Professor. She has been a senior IEEE member since 2010. She has expertise in artificial intelligence focusing on machine learning and computer vision. She has established and sustained a nationally recognized research program to solve challenging research problems. She has acquired competitive funding from reputable agencies and has published 97 high-quality manuscripts. She has worked as a PI on more than 20 research projects and has supervised 82 students. She has also gained her professional leadership experience by serving on technical program committees, NSF panels, and as a reviewer.