

Article

A Labeled GM-PHD Filter for Explicitly Tracking Multiple Targets [†]

Yiyue Gao ¹, Defu Jiang ^{2,*}, Chao Zhang ² and Su Guo ¹

¹ College of Energy and Electrical Engineering, Hohai University, Nanjing 210098, China; gyiyue@hhu.edu.cn (Y.G.); 20060067@hhu.edu.cn (S.G.)

² Laboratory of Array and Information Processing, College of Computer and Information, Hohai University, Nanjing 210098, China; zhangchao2002@hhu.edu.cn

* Correspondence: surfer_jiangdf0801@163.com

[†] This paper is an extension of our conference paper Gao, Y.; Fang, N.; Jiang, D.; Fu, W.; Guo, S. An explicit track continuity algorithm for the GM-PHD filter. In Proceedings of the 31th Chinese Control and Decision Conference, Nanchang, China, 3–5 June 2019.

Abstract: In this study, an explicit track continuity algorithm is proposed for multitarget tracking (MTT) based on the Gaussian mixture (GM) implementation of the probability hypothesis density (PHD) filter. Trajectory maintenance and multitarget state extraction in the GM-PHD filter have not been effectively integrated to date. To address this problem, we propose an improved GM-PHD filter. In this approach, the Gaussian components are classified and labeled, and multitarget state extraction is converted into multiple single-state extractions. This provides the identity label of the individual target and can shield against the negative effects of clutter in the prior density region on the estimates, thus realizing the integration of trajectory maintenance with state extraction in the GM-PHD filter. As no additional associated procedures are required, the overall real-time performance of the proposed filter is similar to or slightly lower than that of the basic GM-PHD filter. The results of numerical experiments demonstrate that the proposed approach can achieve explicit track continuity.



Citation: Gao, Y.; Jiang, D.; Zhang, C.; Guo, S. A Labeled GM-PHD Filter for Explicitly Tracking Multiple Targets. *Sensors* **2021**, *21*, 3932. <https://doi.org/10.3390/s21113932>

Keywords: multitarget tracking; probability hypothesis density filter; Gaussian mixture; track continuity; state extraction

Academic Editor: Gemine Vivone

Received: 24 April 2021

Accepted: 3 June 2021

Published: 7 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background and Motivation

Multitarget tracking (MTT) jointly estimates the target state and the number of targets simultaneously from a sequence of measurements [1]. In MTT, the time-varying number of targets causes data associations between state and measurement sets of targets, which are uncertain. The key challenges encountered involve detection uncertainty, clutter, and data association uncertainty. Therefore, accurately extracting the states of targets and their trajectories has become a critical issue and a hot topic in the field of MTT. The most popular technologies for MTT are the joint probabilistic data association (JPDA) [2,3], multiple hypothesis tracking (MHT) [4], and random finite set (RFS) [1,5].

Since RFS avoids the traditional data association, it has attracted significant attention. Based on RFS and under the framework of Bayesian filtering, many approximations of Bayes multitarget filters have been proposed, mainly including a probability hypothesis density (PHD) filter [1,6,7], cardinalized PHD (CPHD) filter [8], and multi-Bernoulli (MeM-Ber) filter [9]. These filters are not multitarget trackers, which only estimate target states at individual time instants as opposed to multitarget trajectories. While these filters were not designed with the aim of estimating the trajectories of targets [10], they have been used in many applications [11–13].

To provide the information of each track, the notion of labeled RFS is introduced in [14,15]. The generalized labeled multi-Bernoulli (GLMB) filter [14,15] leads to an analytic

solution to the Bayes multitarget tracker, which significantly improves the accuracy of multitarget state extraction and explicitly accommodates the estimation of target trajectories. Recently, labeled RFS-based filters and smoothers [10,16–19] have been further developed, for generating track estimates, which is also the focus of this paper.

A Gibbs-GLMB filter [10] is a computational efficient implementation of the GLMB filter that combines the prediction and update into a single step. A forward-backward labeled multi-Bernoulli smoother in [17] is proposed for MTT, which can improve both the cardinality estimation and the state estimation, and the major computational complexity is linear with the number of targets. The trajectory Poisson multi-Bernoulli filter and the trajectory Poisson multi-Bernoulli mixture filter [18] have better filtering accuracy and real-time performance than those of the Gibbs-GLMB filter. A onetime step lagged Bayes multitarget smoother using the δ -GLMB density [19], which also inherently outputs targets trajectories, outperforms the Gibbs-GLMB filter on both the estimates and target number and state at the cost of higher computational complexity. However, when targets are in a dense clutter, or misdetected, generating the correct multitarget trajectories is difficult in [16–19], although the computational complexity problem has been improved in [17,18].

Thus, it is necessary to design a relatively low computing cost filter that can provide explicit trajectories whilst considering the intense clutter environment.

1.2. Brief Survey of Related Work

From the viewpoint of practical applicability, the PHD filter is suitable for applications demanding real-time results [20–22]. Unlike the full Bayesian filter, the PHD filter iteratively propagates the first moment of the multitarget posterior density. It is a promising technique in terms of computational complexity to solve the MTT problem. However, it does not explicitly accommodate the trajectories of targets [10,21,23]. Furthermore, the PHD filter is known to produce highly uncertain estimates of the target number [21,23]. In recent years, several modifications have been proposed for the trajectories of targets [22,24–30].

In [29], an explicit track continuity algorithm for the sequence Monte Carlo PHD (SMC-PHD) filter [31] is proposed, which not only shields against the negative effects of clutter but provides the identities of the individual targets. However, when several newborn targets appear simultaneously in one newborn region, it is unable to extract the state estimates of different observations from these newborn targets. Compared to the SMC-PHD filter, the Gaussian mixture PHD (GM-PHD) filter [32] has the advantages of simple state extraction and low computational cost, which is more suitable for the requirement of real-time scenes. Thus, the focus of this study is the GM-PHD filter.

For the estimation of the target trajectory of the GM-PHD filter, recently proposed solutions involve integrating the track identity into the GM-PHD filter. In each iteration process, each Gaussian component is assigned a label to indicate its correlation with one track. These methods can be mainly classified into two groups. One group [24,25] involves running an additional association method based on the estimates outputted by the filter. The other solution [26–29] extends a label to the state estimate. However, when targets are in close proximity to each other, in a dense clutter, or misdetected, maintaining the correct multitarget trajectories is difficult, although the accuracy of the estimates of the target number has been increased significantly by improving multitarget state extraction strategies [33–37]. Thus, one can conclude that in the GM-PHD filter, track maintenance and multitarget state extraction have not been effectively integrated. Recently, a Gaussian mixture trajectory PHD filter was proposed in [30], which is able to provide trajectory estimates for live targets, without adding labels or tags. It adapts the estimator for the GMPHD filter for the sets of trajectories; thus, its tracking accuracy is the same as that of the PHD filter. When clutters appear in the high prior density region of the target, it is difficult to shield against the interference of clutters.

An instructive idea is that the Gaussian components approximating the posterior information of one target have the same identity label, which will not be changed throughout

the overall filtering process. In the multitarget state extraction, only one state estimate can be extracted from the updated survival Gaussian components with the same label. Then, the explicit trajectories can be obtained by simply connecting the state estimates with the same label, which requires no additional track management. To achieve this concept, in this paper, we propose an explicit track continuity algorithm for the GM-PHD filter.

1.3. Main Contributions

The work in this paper is based on a published conference paper [38]. The major contributions of this paper are as follows:

- First, each Gaussian component is assigned a label, and the identity labels of Gaussian components are divided into three classes; once it is determined as a confirmed component, its label will be unchanged throughout the whole filtering process. This is convenient for track maintenance; that is, state estimates with the same label at different times belong to the same track.
- Second, based on the measurements selected by eliminating most clutter, the updated components are obtained, and their parameters are stored in four different query tables. Based only on these query tables, the multitarget state extraction is converted to multiple single-target state extractions providing the identity label, and the negative effects of clutter in the prior density region on the estimates can be prevented. Thus, we realize the integration of trajectory maintenance with state extraction in the GM-PHD filter.
- Third, since the component updating process is based on the selected measurements and no additional associated process is required, the overall real-time performance of the proposed filter is similar to or slightly lower than that of the basic GM-PHD filter. The performance of the proposed approach is verified by simulations designed in a linear scenario.

The remainder of this paper is organized as follows. The technical background is provided in Section 2, while the specific designing of the improved GM-PHD filter is presented in Section 3. The quantitative experiments are described in Section 4, and concluding remarks are provided in Section 5.

2. Background

2.1. GM-PHD Filter

In MTT, the state and observation sets are time-varying and unknown. The collection of the states of targets at time k is represented as an RFS $X_k = \{x_k^{(1)}, \dots, x_k^{(i)}, \dots, x_k^{(|X_k|)}\}$, where $|X_k|$ denotes the number of targets, and $x_k^{(i)}$ denotes the state of an individual target. Similarly, the collection of available measurements is an RFS $Z_k = \{z_k^{(1)}, \dots, z_k^{(i)}, \dots, z_k^{(|Z_k|)}\}$, where $|Z_k|$ is the number of measurements and $z_k^{(i)}$ is a measurement from a target or due to a clutter. The PHD [6] is the first-order moment of multitarget posterior density.

Assuming that each target follows a linear Gaussian dynamical model and the sensor has a linear Gaussian measurement model [32],

$$f_{k|k-1}(x_k | x_{k-1}) = \mathcal{N}(x_k; F_{k-1}x_{k-1}, Q_{k-1}) \quad (1)$$

$$g_k(z_k | x_k) = \mathcal{N}(z_k; H_k x_k, R_k) \quad (2)$$

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

The birth intensity function of the new targets at time k [32] is $\gamma_k(x) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}(x; m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)})$, where $J_{\gamma,k}$ denotes the number of newborn Gaussian components, and $w_{\gamma,k}^{(i)}$, $m_{\gamma,k}^{(i)}$, and $P_{\gamma,k}^{(i)}$ are the weight, mean, and covariance, respectively, of the i th newborn

Gaussian component. Suppose that the survival probability $p_{S,k} = p_{S,k}(\mathbf{x}_k)$, the detection probability $p_{D,k} = p_{D,k}(\mathbf{x}_k)$. Based on these assumptions [32], the PHD filter can be efficiently approximated by mixed Gaussian components. w is the weight of Gaussian density and the posterior intensity $D_{k-1|k-1}(\mathbf{x})$ at time $k-1$ [32] is a Gaussian mixture form with J_{k-1} survival components as follows:

$$D_{k-1|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k-1}^{(i)}, \mathbf{P}_{k-1}^{(i)}) \quad (3)$$

The predicted intensity $D_{k|k-1}(\mathbf{x})$ at time k is also a Gaussian mixture with $J_{k|k-1}$ components calculated as follows:

$$\begin{aligned} D_{k|k-1}(\mathbf{x}) &= \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)}) \\ &= D_{S,k|k-1}(\mathbf{x}) + \gamma_k(\mathbf{x}) \end{aligned} \quad (4)$$

where $D_{S,k|k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} w_{S,k|k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{S,k|k-1}^{(i)}, \mathbf{P}_{S,k|k-1}^{(i)})$ is the intensity of survival targets, $w_{S,k|k-1}^{(i)} = p_{S,k} w_{k-1}^{(i)}$, $\mathbf{m}_{S,k|k-1}^{(i)} = \mathbf{F}_{k-1} \mathbf{m}_{k-1}^{(i)}$, $\mathbf{P}_{S,k|k-1}^{(i)} = \mathbf{F}_{k-1} \mathbf{P}_{k-1}^{(i)} (\mathbf{F}_{k-1})^T + \mathbf{Q}_{k-1}$.

When a new set Z_k arrives, the posterior intensity $D_{k|k}$ at time k [32] is updated and can be described as follows:

$$D_{k|k}(\mathbf{x}) = (1 - p_{D,k}) D_{k|k-1}(\mathbf{x}) + \sum_{z \in Z_k} D_{D,k}(\mathbf{x}; z) \quad (5)$$

where

$$D_{D,k}(\mathbf{x}; z) = \sum_{i=1}^{J_{k|k-1}} w_{k|k}^{(i)}(z) \mathcal{N}(\mathbf{x}; \mathbf{m}_{k|k}^{(i)}(z), \mathbf{P}_{k|k}^{(i)}) \quad (6)$$

$$w_{k|k}^{(i)}(z) = \frac{p_{D,k} w_{k|k-1}^{(i)} g_k(z | \mathbf{m}_{k|k-1}^{(i)})}{\lambda c(z) + p_{D,k} \sum_{l=1}^{J_{k|k-1}} w_{k|k-1}^{(l)} g_k(z | \mathbf{m}_{k|k-1}^{(l)})} \quad (7)$$

$$g_k(z | \mathbf{m}_{k|k-1}^{(i)}) = \mathcal{N}(z; \mathbf{H}_k \mathbf{m}_{k|k-1}^{(i)}, \mathbf{S}_k^{(i)}) \quad (8)$$

$$\mathbf{m}_{k|k}^{(i)}(z) = \mathbf{m}_{k|k-1}^{(i)} + \mathbf{K}_k^{(i)} (z - \mathbf{H}_k \mathbf{m}_{k|k-1}^{(i)}) \quad (9)$$

$$\mathbf{P}_{k|k}^{(i)} = [\mathbf{I} - \mathbf{K}_k^{(i)} \mathbf{H}_k] \mathbf{P}_{k|k-1}^{(i)} \quad (10)$$

$$\mathbf{S}_k^{(i)} = \mathbf{R}_k + \mathbf{H}_k \mathbf{P}_{k|k-1}^{(i)} (\mathbf{H}_k)^T \quad (11)$$

$$\mathbf{K}_k^{(i)} = \mathbf{P}_{k|k-1}^{(i)} (\mathbf{H}_k)^T (\mathbf{H}_k \mathbf{P}_{k|k-1}^{(i)} (\mathbf{H}_k)^T + \mathbf{R}_k)^{-1} \quad (12)$$

For more details on the GM-PHD filter, we refer the reader to the original study [32].

To reduce the computational load, the increasing Gaussian components must be pruned and merged. Then, the posterior intensity $D_{k|k}$ is represented as follows:

$$D_{k|k}(\mathbf{x}) = \sum_{i=1}^{J_k} w_k^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_k^{(i)}, \mathbf{P}_k^{(i)}) \quad (13)$$

The multiple-target state estimates can be extracted based on the threshold w_{th} (this is generally 0.5) from $D_{k|k}(\mathbf{x})$. Figure 1 shows the overall procedure of the basic GM-PHD filter.

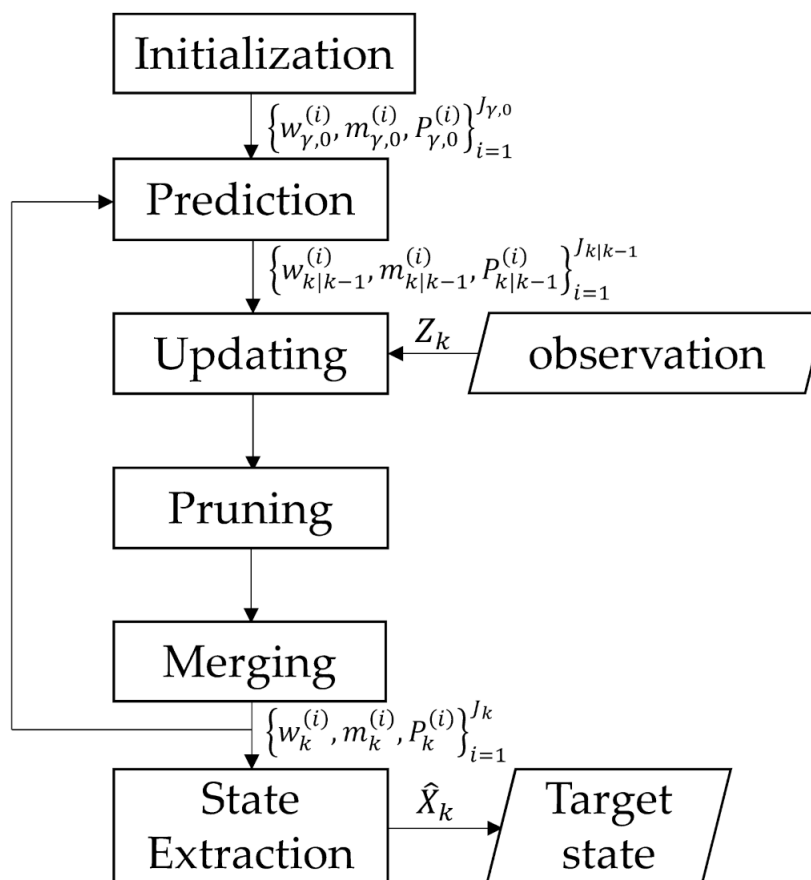


Figure 1. Overall procedure of the basic GM-PHD filter.

2.2. Problem Formulation

The basic GM-PHD filter in Figure 1 does not explicitly accommodate target trajectory estimation. Trajectory continuity and state estimates essentially influence and interact with each other. In MTT, detection uncertainty, false state estimates, and closely spaced targets have detrimental effects on track continuity. Figure 2 graphically depicts these phenomena.

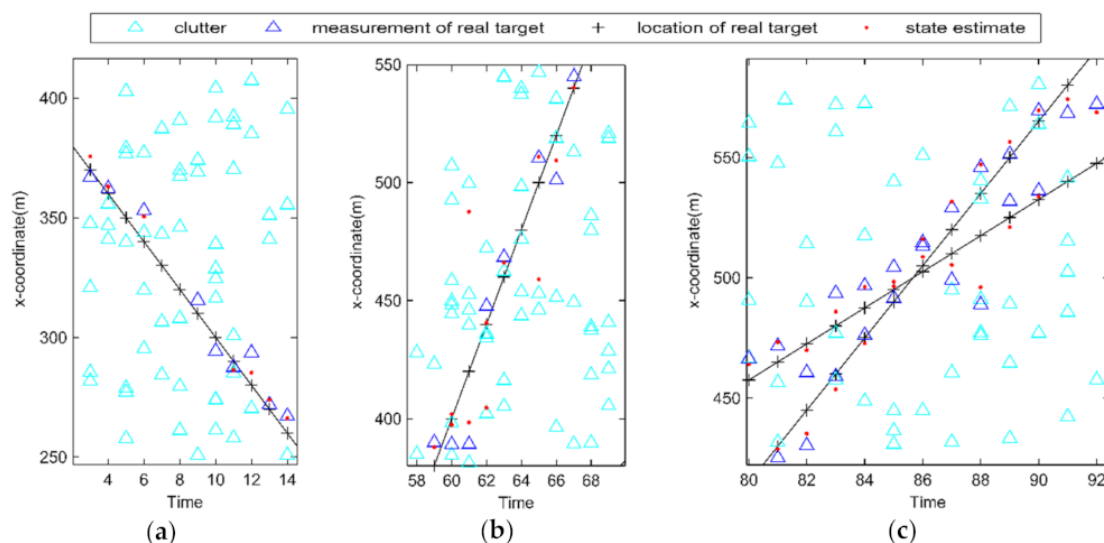


Figure 2. Phenomena affecting track continuity of the basic GM-PHD filter. (a) detection uncertainty; (b) false state estimates; (c) closely spaced targets.

As shown in Figure 2, the clutter rate was relatively high, and the detection probability was 0.9 in the current tracking scenario. In Figure 2a, at times $k = 7$ and $k = 8$, the target was misdetections, and its posterior intensity was weakened. This caused its inability to obtain timely state estimates even if the target was redetected at times $k = 9$ and $k = 10$. Managing the trajectories of the state estimates at times $k = 6$ and $k = 11$ was a challenge. In Figure 2b, at times $k = 60$, $k = 61$, $k = 62$, and $k = 65$, there were false state estimates caused by clutter points appearing in the high-prior density region of the target. In Figure 2c, two targets were closely spaced, and their state estimates were nearby. It is difficult to identify the state estimate with which they should be associated. The above phenomena make it difficult to provide a correct target trajectory based on the basic GM-PHD filter.

A trajectory is a time-sequence of state estimates with the same label, such as $\{(\hat{x}_{k-1}, \ell), (\hat{x}_k, \ell), (\hat{x}_{k+2}, \ell)\}$, wherein ℓ provides the means to identify the trajectory or track of one target. Furthermore, even if there is no state estimate at time $k + 1$, the correct track continuity can still be obtained. To achieve explicit trajectories based on the GM-PHD filter, we proposed an algorithm that labels all Gaussian components, which we called the labeled GM-PHD (LGM-PHD) filter.

3. The Proposed Multitarget Tracking Algorithm

Assuming the target birth intensity is known a priori, a “one-to-one” principle for the multitarget state extraction of the GM-PHD filter was proposed herein.

The “one-to-one” principle for the GM-PHD filter: In the multitarget state extraction, only one state estimate can be extracted from the updated Gaussian components with the same label, originating from the survival ones.

To abide by the “one-to-one” principle, each Gaussian component is assigned a label, and these labels are divided into three categories in the LGM-PHD filter. In this paper, the identity labels of the Gaussian components are divided into three classes, the labels of newborn Gaussian components, the labels of unconfirmed Gaussian components (approximating the posterior information of targets having their own unconfirmed trajectory), and the labels of confirmed components (approximating the posterior information of targets having their own confirmed trajectory). The attribute of a label ℓ is determined based on the value range as follows:

$$\begin{aligned} & \text{label of a newborn component, if } \ell \geq V_{new}, \\ & \text{label of an unconfirmed component, if } V_{un} \leq \ell < V_{new}, \\ & \text{label of a confirmed component, if } 0 \leq \ell < V_{un}. \end{aligned} \quad (14)$$

where V_{un} is a value far greater than the total number of targets that could possibly appear in this scenario, $V_{new} \neq V_{un}$, and $V_{new} \gg V_{un}$. In general, $V_{un} = 200$ and $V_{new} = 400$.

r_{max} and r_{maxf} are used to count the confirmed and unconfirmed trajectories, respectively. At time $k = 0$, r_{max} is initialized as 0 and r_{maxf} is initialized as V_{un} . When a new confirmed trajectory is initiated, r_{max} is incremented by one; when a new unconfirmed trajectory is initiated, r_{maxf} is also incremented by one.

The overall procedure of the LGM-PHD filter contains seven steps—initialization, prediction, observation selection, updating, state extraction, pruning, and merging and capping—as shown in Figure 3. When compared with the basic GM-PHD filter in Figure 1, the submodules marked with “*” are either modified or proposed in this paper.

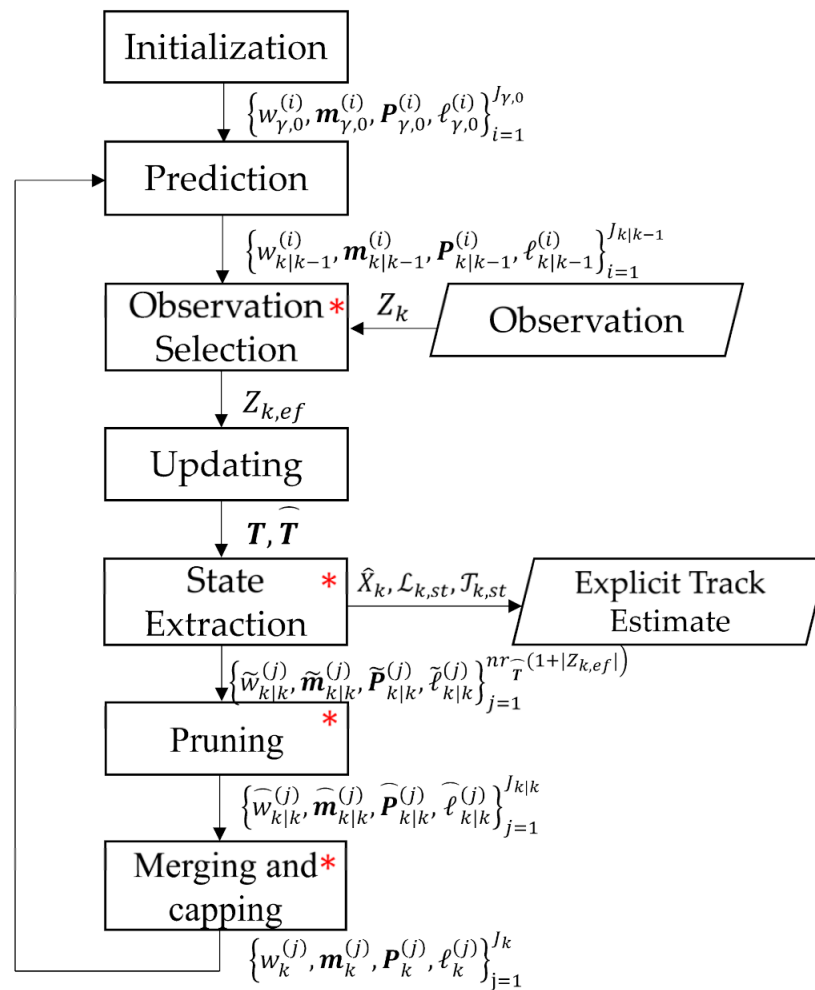


Figure 3. Overall procedure of the LGM-PHD filter.

3.1. Initialization and Prediction

At time k , newborn targets appear spontaneously according to a Poisson point process with the intensity function

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^{(i)}, \mathbf{P}_{\gamma,k}^{(i)}) \quad (15)$$

where each newborn Gaussian component is assigned a label $\ell_{\gamma,k}^{(i)} = V_{new} + i, i = 1, \dots, J_{\gamma,k}$. According to (15), the initial intensity $D_{0|0}$ is described as follows: $\{w_{\gamma,0}^{(i)}, \mathbf{m}_{\gamma,0}^{(i)}, \mathbf{P}_{\gamma,0}^{(i)}, \ell_{\gamma,0}^{(i)}\}_{i=1}^{J_{\gamma,0}}$.

With this assumption, at time $k - 1$, the posterior probability density $D_{k-1|k-1}(\mathbf{x})$ is approximated by $\{w_{k-1}^{(j)}, \mathbf{m}_{k-1}^{(j)}, \mathbf{P}_{k-1}^{(j)}, \ell_{k-1}^{(j)}\}_{j=1}^{J_{k-1}}$, where $\ell_{k-1}^{(j)}$ is the identity label of the j th survival Gaussian component. Then, according to (4), the priori density at time k is approximated by $\{w_{k|k-1}^{(i)}, \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)}, \ell_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}}$, where $J_{k|k-1} = J_{k-1} + J_{\gamma,k}$. The i th predicted Gaussian component is labeled as follows:

$$\ell_{k|k-1}^{(i)} = \begin{cases} \ell_{k-1}^{(i)}, & i = 1, \dots, J_{k-1} \\ \ell_{\gamma,k}^{(i-J_{k-1})}, & i = J_{k-1} + 1, \dots, J_{k|k-1} \end{cases} \quad (16)$$

To obtain the time-sequence of states with the same label, the labels of conformed components remain unchanged throughout the overall filtering process. In addition, the

labels of all types of components can be inherited, and only components with the same label can be merged.

3.2. Observation Selection

To achieve robust track maintenance, clutter should be eliminated. Considering the overall real-time performance of the filter, it is necessary to eliminate most clutter before the updating procedure, to select effective observations. Given the zero-mean-measurement Gaussian white noise with the covariance matrix $\mathbf{R}_k = \text{diag}([\sigma_x^2 \ \sigma_y^2]^T)$, the threshold $d(a)$ is calculated by $d(a) = a\|[\sigma_x \ \sigma_y]^T - [0 \ 0]^T\|$, in which $\|\cdot\|$ is the Euclidean distance and a is the confidence coefficient. In this gating algorithm, $a = 6$. Initializing the effective observation set $Z_{k,ef}$ as \emptyset , based on all the predicted Gaussian components at time k , $Z_{k,ef} = \{z_{k,ef}^{(j)}\}_{j=1}^{|Z_{k,ef}|}$ is obtained by the operations in Algorithm 1.

Algorithm 1. The observation selection for the GM-PHD filter algorithm

```

1: Given  $\{w_{k|k-1}^{(i)}, m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}\}_{i=1}^{J_{k|k-1}}$  and  $Z_k = \{z_k^{(j)}\}_{j=1}^{|Z_k|}$ 
2: Initialization  $Z_{k,ef} = \emptyset$ 
3: for  $j = 1, \dots, |Z_k|$  do
4:   for  $i = 1, \dots, J_{k|k-1}$  do
5:      $-d^{(j,i)} = \|z_k^{(j)} - H_k m_{k|k-1}^{(i)}\|$ 
6:     -if  $d^{(j,i)} \leq d(a)$ 
7:        $-Z_{k,ef} = [Z_{k,ef}, z_k^{(j)}]$ 
8:     -break
9:   -end
10: end
11: end

```

3.3. Updating

First, four tables \mathbf{U}_w , \mathbf{U}_m , \mathbf{U}_p , and \mathbf{U}_ℓ , all with size $J_{k|k-1} \times (1 + |Z_{k,ef}|)$, are created to store parameters w , m , P , and ℓ of each updated Gaussian component. They are uniformly called query table \mathbf{U} .

Second, based on $Z_{k,ef}$, the posterior intensity $D_{k|k}$ at time k can be described as follows: $D_{k|k}(\mathbf{x}) = (1 - p_{D,k})D_{k|k-1}(\mathbf{x}) + \sum_{z_{k,ef}^{(j)} \in Z_{k,ef}} D_{D,k}(\mathbf{x}; z_{k,ef}^{(j)})$, where $D_{D,k}(\mathbf{x}; z_{k,ef}^{(j)}) =$

$\sum_{i=1}^{J_{k|k-1}} w_{k|k}^{(i)}(z_{k,ef}^{(j)}) \mathcal{N}(\mathbf{x}; m_{k|k}^{(i)}(z_{k,ef}^{(j)}), P_{k|k}^{(i)})$. Based on the i th predicted Gaussian component and $z_{k,ef}^{(j)}$, we obtain the parameter set $\{w_{k|k}^{(i)}(z_{k,ef}^{(j)}), m_{k|k}^{(i)}(z_{k,ef}^{(j)}), P_{k|k}^{(i)}, \ell_{k|k}^{(i)}\}$ of the updated Gaussian component via (5)–(12), where $\ell_{k|k}^{(i)} = \ell_{k|k-1}^{(i)}$. Based on this parameter set, the elements in the i th row and the $(j + 1)$ th column of \mathbf{U}_w , \mathbf{U}_m , \mathbf{U}_p , and \mathbf{U}_ℓ are obtained as follows:

$$\begin{aligned}
 \mathbf{U}_w(i, j + 1) &= w_{k|k}^{(i)}(z_{k,ef}^{(j)}) \\
 \mathbf{U}_m(i, j + 1) &= m_{k|k}^{(i)}(z_{k,ef}^{(j)}) \\
 \mathbf{U}_p(i, j + 1) &= P_{k|k}^{(i)} \\
 \mathbf{U}_\ell(i, j + 1) &= \ell_{k|k}^{(i)}
 \end{aligned} \tag{17}$$

$(1 - p_{D,k})D_{k|k-1}(\mathbf{x})$ is approximated by the parameter set $\left\{ (1 - p_{D,k})w_{k|k-1}^{(i)}, \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)}, \ell_{k|k-1}^{(i)} \right\}_{i=1}^{J_{k|k-1}}$. Based on this parameter set, the elements in the i th row and the 1th column of \mathbf{U}_w , \mathbf{U}_m , \mathbf{U}_P , and \mathbf{U}_ℓ are obtained as follows:

$$\begin{aligned} \mathbf{U}_w(i, 1) &= (1 - p_{D,k})w_{k|k-1}^{(i)} \\ \mathbf{U}_m(i, 1) &= \mathbf{m}_{k|k-1}^{(i)} \\ \mathbf{U}_P(i, 1) &= \mathbf{P}_{k|k-1}^{(i)} \\ \mathbf{U}_\ell(i, 1) &= \ell_{k|k-1}^{(i)} \end{aligned} \quad (18)$$

Third, four tables, $\widehat{\mathbf{U}}_w$, $\widehat{\mathbf{U}}_m$, $\widehat{\mathbf{U}}_P$, and $\widehat{\mathbf{U}}_\ell$, are created, which are collectively called the $\widehat{\mathbf{U}}$

$$\widehat{\mathbf{U}}_w = \mathbf{U}_w, \widehat{\mathbf{U}}_m = \mathbf{U}_m, \widehat{\mathbf{U}}_P = \mathbf{U}_P, \widehat{\mathbf{U}}_\ell = \mathbf{U}_\ell \quad (19)$$

When a new confirmed trajectory or a new unconfirmed trajectory is established, its information is documented and stored in $\widehat{\mathbf{U}}$.

3.4. State Extraction

In this section, by means of \mathbf{U} , certain Gaussian components with the same label are associated with one effective observation, and if the maximum weight in these components satisfies some conditions, one state estimate is extracted.

Complete target status information contains state estimate $\hat{\mathbf{x}}$, track label ℓ , and occurrence time k . Thus, at time k , the state estimate set \hat{X}_k , the track label set $\mathcal{L}_{k,st}$, and the occurrence time set $\mathcal{T}_{k,st}$ are all initialized as \emptyset . The specific steps in state extraction process are as follows.

Step 1: First, obtain the maximum value \bar{w} in $\mathbf{U}_w(:, 2 : end)$, where $\mathbf{U}_w(:, 2 : end)$ represents the elements from the second column to the last column of each row in \mathbf{U}_w . $\mathbf{U}_w(i, j)$ denotes the element in the j th column of the i th row in \mathbf{U}_w . Here, $\bar{w} = \mathbf{U}_w(i^*, j^*)$ is the maximum, $(i^*, j^*) = \operatorname{argmax}(\mathbf{U}_w(i, j))$, $\forall i = 1, \dots, J_{k|k-1}$, and $\forall j = 2, \dots, 1 + |Z_{k,ef}|$. Very rarely, \bar{w} may be in two or more different locations in \mathbf{U}_w .

Next, if $\bar{w} < w_b$, jump to Step 4. w_b is the threshold of the basic weight. In general, w_b is set to 0.02. If $\bar{w} \geq w_b$, based on the row and column numbers of \bar{w} , obtain the unique label set $\mathcal{L}_{\bar{w}} = \left\{ \ell_{\bar{w}}^{(i)} = \mathbf{U}_\ell(rn^{(i)}, cn^{(i)}) \right\}_{i=1}^{|\mathcal{L}_{\bar{w}}|}$, and initialize $CN = \emptyset$. CN is used to store the column numbers of observations based on which some tracks will be established or maintained. In most cases, the label set only has one element, i.e., $|\mathcal{L}_{\bar{w}}| = 1$.

Step 2: For $i = 1, \dots, |\mathcal{L}_{\bar{w}}|$, determine the attribute of $\ell_{\bar{w}}^{(i)}$, and execute the following actions for the Gaussian components with $\ell_{\bar{w}}^{(i)}$.

- First, obtain $\hat{\mathbf{x}}$ by $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(rn^{(i)}, cn^{(i)})$. The row numbers of elements with $\ell_{\bar{w}}^{(i)}$ in \mathbf{U}_ℓ are denoted as $RN_{\ell_{\bar{w}}^{(i)}}$.
- Next, the attribute of $\ell_{\bar{w}}^{(i)}$ will be used to determine which of the following three cases will be executed.

Case A: $\ell_{\bar{w}}^{(i)} \geq V_{new}$, the label of newborn components.

As several newborn targets may appear simultaneously in one newborn region, newborn Gaussian components with the same label may contain information on several newborn targets. Therefore, several components with large weights in $\mathbf{U}_w \left(RN_{\ell_{\bar{w}}^{(i)}}, 2 : (1 + |Z_{k,ef}|) \right)$ can be extracted as the state estimates of different observations, and new tracks can be established accordingly. The specific steps are as follows:

① The sequence number set CN_{w_l} of the different observations with large weights in $\mathbf{U}_w \left(RN_{\ell_{\overline{w}}^{(i)}}, 2 : (1 + |Z_{k,ef}|) \right)$ is obtained as follows:

$$\begin{aligned} CN'_{w_l} &= \left\{ j \mid \mathbf{U}_w \left(RN_{\ell_{\overline{w}}^{(i)}}, j \right) \geq w_l, \forall j = 2, \dots, 1 + |Z_{k,ef}| \right\} \\ CN_{w_l} &= \left\{ cn_{w_l}^{(j1)} \right\}_{j1=1}^{|CN'_{w_l}|} = uni(CN'_{w_l}) \end{aligned} \quad (20)$$

where w_l is the threshold of large weights and $uni(A)$ is an operator that obtains the unique elements in set A . In general, w_l is set to 0.4.

Additionally, $CN = [CN, CN_{w_l}]$.

② If $|CN_{w_l}| \geq 1$, for $j1 = 1, \dots, |CN_{w_l}|$, $|CN_{w_l}|$ state estimates will be extracted for the large weight observations and $|CN_{w_l}|$ new confirmed tracks will be established. The $j1$ th target status information is obtained as follows:

$$\begin{aligned} r_{\max} &= r_{\max} + 1, \\ rn_{w_l} &= \arg \max_{i1 \in RN_{\ell_{\overline{w}}^{(i)}}} \left(\mathbf{U}_w \left(i1, cn_{w_l}^{(j1)} \right) \right), \\ \hat{X}_k &= \left[\hat{X}_k, \mathbf{H}_k \mathbf{U}_m \left(rn_{w_l}, cn_{w_l}^{(j1)} \right) \right], \\ \mathcal{L}_{k,st} &= [\mathcal{L}_{k,st}, r_{\max}], \\ \mathcal{T}_{k,st} &= [\mathcal{T}_{k,st}, k]. \end{aligned} \quad (21)$$

In (21), the label of the new confirmed track is given by $r_{\max} = r_{\max} + 1$, rn_{w_l} is the row number of the maximum weight in $\mathbf{U}_w \left(RN_{\ell_{\overline{w}}^{(i)}}, cn_{w_l}^{(j1)} \right)$ and the single-target state estimate is $\mathbf{H}_k \mathbf{U}_m \left(rn_{w_l}, cn_{w_l}^{(j1)} \right)$, which is stored in state estimate set \hat{X}_k and labeled as r_{\max} . The information on the $j1$ th new confirmed track is established by modifying $\widehat{\mathbf{U}}_w$, $\widehat{\mathbf{U}}_m$, $\widehat{\mathbf{U}}_p$, and $\widehat{\mathbf{U}}_\ell$ as follows:

$$\left. \begin{aligned} \widehat{\mathbf{U}}_w \left(RN_{\ell_{\overline{w}}^{(i)}}, : \right) &= 0, \\ \widehat{\mathbf{U}}_w \left(rn_{w_l}, cn_{w_l}^{(j1)} \right) &= \mathbf{U}_w \left(rn_{w_l}, cn_{w_l}^{(j1)} \right), \\ \widehat{\mathbf{U}}_\ell \left(RN_{\ell_{\overline{w}}^{(i)}}, : \right) &= r_{\max}. \end{aligned} \right\} \text{if } j1 = 1. \quad (22)$$

$$\left. \begin{aligned} \widehat{\mathbf{U}}_w(\text{end} + 1, :) &= 0, \\ \widehat{\mathbf{U}}_w(\text{end}, cn_{w_l}^{(j1)}) &= \mathbf{U}_w \left(rn_{w_l}, cn_{w_l}^{(j1)} \right), \\ \widehat{\mathbf{U}}_\ell(\text{end} + 1, :) &= r_{\max}, \\ \widehat{\mathbf{U}}_m(\text{end} + 1, :) &= \mathbf{U}_m \left(rn_{w_l}, : \right), \\ \widehat{\mathbf{U}}_p(\text{end} + 1, :) &= \mathbf{U}_p \left(rn_{w_l}, : \right). \end{aligned} \right\} \text{if } j1 > 1.$$

where $\widehat{\mathbf{U}}(\text{end}, j)$ denotes the elements in the j th column of the last row in $\widehat{\mathbf{U}}$, and $\widehat{\mathbf{U}}(\text{end} + 1, :)$ denotes adding one row at the bottom of $\widehat{\mathbf{U}}$. If $j1 > 1$, it means at least one state estimate has been extracted at time k . In other words, two or more confirmed trajectories will be simultaneously established from one prior newborn region. Thus, one additional row should be added in $\widehat{\mathbf{U}}$ to establish the posterior information of the individual track. To shield against interference from closely spaced components with another label, only the elements in $\mathbf{U} \left(rn_{w_l}, cn_{w_l}^{(j1)} \right)$ can be reserved.

③ Additionally, some components whose weights are between w_s and w_l in $\mathbf{U}_w \left(RN_{\ell_{\bar{w}}^{(i)}}, 2 : (1 + |Z_{k,ef}|) \right)$ should also be associated with different observations. The sequence number set $CN_{w_{bl}}$ of these observations with small weights is given as follows:

$$\begin{aligned} CN''_{w_{bl}} &= \left\{ j \mid w_s \leq \mathbf{U}_w \left(RN_{\ell_{\bar{w}}^{(i)}}, j \right) < w_l, \forall j = 2, \dots, 1 + |Z_{k,ef}| \right\} \\ CN'_{w_{bl}} &= \text{uni}(CN''_{w_{bl}}) \\ CN_{w_{bl}} &= \left\{ cn_{w_{bl}}^{(j2)} \right\}_{j2=1}^{|CN''_{w_{bl}}|} = CN''_{w_{bl}} - CN_{w_l} \end{aligned} \quad (23)$$

where w_s is the threshold of small weights. In general, w_s is set to 0.1.

Additionally, $CN = [CN, CN_{w_{bl}}]$.

④ If $|CN_{w_{bl}}| \geq 1$, for $j2 = 1, \dots, |CN_{w_{bl}}|$, $|CN_{w_l}|$ unconfirmed tracks will be established for the small weight observations. The information on the $j2$ th unconfirmed track is obtained by specific steps as follows:

$$r_{\max t} = r_{\max t} + 1 \quad (24)$$

$$rn_{w_{bl}} = \arg \max_{i1 \in RN_{\ell_{\bar{w}}^{(i)}}} \left(\mathbf{U}_w \left(i1, cn_{w_{bl}}^{(j2)} \right) \right) \quad (25)$$

$$\left. \begin{aligned} \widehat{\mathbf{U}}_w \left(RN_{\ell_{\bar{w}}^{(i)}}, : \right) &= 0, \\ \widehat{\mathbf{U}}_w \left(rn_{w_{bl}}, cn_{w_{bl}}^{(j2)} \right) &= \mathbf{U}_w \left(rn_{w_{bl}}, cn_{w_{bl}}^{(j2)} \right), \\ \widehat{\mathbf{U}}_\ell \left(RN_{\ell_{\bar{w}}^{(i)}}, : \right) &= r_{\max t}. \end{aligned} \right\} \text{if } |CN_{w_l}| = 0 \& j2 = 1. \quad (26)$$

$$\left. \begin{aligned} \widehat{\mathbf{U}}_w(\text{end} + 1, :) &= 0, \\ \widehat{\mathbf{U}}_w(\text{end}, cn_{w_{bl}}^{(j2)}) &= \mathbf{U}_w(rn_{w_{bl}}, cn_{w_{bl}}^{(j2)}), \\ \widehat{\mathbf{U}}_\ell(\text{end} + 1, :) &= r_{\max t}, \\ \widehat{\mathbf{U}}_m(\text{end} + 1, :) &= \mathbf{U}_m(rn_{w_{bl}}, :), \\ \widehat{\mathbf{U}}_p(\text{end} + 1, :) &= \mathbf{U}_p(rn_{w_{bl}}, :). \end{aligned} \right\} \text{if } (|CN_{w_l}| = 0 \& j2 > 1) \mid (|CN_{w_l}| \geq 1).$$

where $\widehat{\mathbf{U}}(\text{end}, j)$ and $\widehat{\mathbf{U}}(\text{end} + 1, :)$ are the same as those in (22).

Note that no state estimate is extracted here, since state estimates are only extracted from newborn components with large weight or from confirmed components. Storing the parameter set in \mathbf{U} makes it easy to determine the observation associated with one updated component with the maximum weight, such as CN_{w_l} and $CN_{w_{bl}}$.

Case B: $V_{un} \leq \ell_{\bar{w}}^{(i)} < V_{new}$, the label of unconfirmed components.

① If $\bar{w} \geq w_l$ or $(w_m \leq \bar{w} < w_l) \& (\ell_{\bar{w}}^{(i)} \in \mathcal{L}_{k-1,un}) \& (\ell_{\bar{w}}^{(i)} \in \mathcal{L}_{k-2,un})$, new target status information is obtained as follows:

$$\begin{aligned} r_{\max} &= r_{\max} + 1, \\ \hat{\mathbf{X}}_k &= \left[\hat{\mathbf{X}}_k, \mathbf{H}_k \mathbf{U}_m(rn^{(i)}, cn^{(i)}) \right], \\ \mathcal{L}_{k,st} &= [\mathcal{L}_{k,st}, r_{\max}], \\ \mathcal{T}_{k,st} &= [\mathcal{T}_{k,st}, k]. \end{aligned} \quad (27)$$

where $\mathcal{L}_{k-1,un}$ is the label set of unconfirmed components with small weights at time $k - 1$, $w_b < w_m < w_l$. In general, w_m is set to 0.2.

The information on this new confirmed track is established by modifying $\widehat{\mathbf{U}}_w$ and $\widehat{\mathbf{U}}_\ell$ as follows:

$$\begin{aligned}\widehat{\mathbf{U}}_w\left(RN_{\ell_{\bar{w}}^{(i)}},:\right) &= 0, \\ \widehat{\mathbf{U}}_w\left(RN_{\ell_{\bar{w}}^{(i)}},cn^{(i)}\right) &= \mathbf{U}_w\left(RN_{\ell_{\bar{w}}^{(i)}},cn^{(i)}\right), \\ \widehat{\mathbf{U}}_\ell\left(RN_{\ell_{\bar{w}}^{(i)}},:\right) &= r_{\max}.\end{aligned}\quad (28)$$

Additionally, $CN = [CN, cn^{(i)}]$.

② Otherwise, $\ell_{\bar{w}}^{(i)}$ is stored in $\mathcal{L}_{k,un}$, i.e., $\mathcal{L}_{k,un} = [\mathcal{L}_{k,un}, \ell_{\bar{w}}^{(i)}]$.

Case C: $0 \leq \ell_{\bar{w}}^{(i)} < V_{un}$, the label of confirmed components. The single-target state estimate is extracted.

① Preprocessing state estimate. If the observation of one target drifts from its real location, or one target is misdetection at time $k - 1$, the distribution of this target's posterior information will diverge, and the weights of components associated with its observation at time k will be weakened. Thus, if $\bar{w} < w_m$, $rn_{w_m} = \arg \max_{i1 \in RN_{\ell_{\bar{w}}^{(i)}}}(\mathbf{U}_w(i1, 1))$. Furthermore, if

$\|\mathbf{H}_k \mathbf{U}_m(rn^{(i)}, cn^{(i)}) - \mathbf{H}_k \mathbf{U}_m(rn_{w_m}, 1)\| > d(1)$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(rn_{w_m}, 1)$. The definition of $d(\cdot)$ is provided in Section 3.2.

Remark 1. As shown in Figure A1, the 1st column elements in \mathbf{U} represent the updated information for misdetection at time k . When there is some deviation, i.e., $\bar{w} < w_m$, $\mathbf{H}_k \mathbf{U}_m(rn_{w_m}, 1)$ is the optimal state estimate since $\mathbf{U}_m(rn_{w_m}, 1)$ is the update of the posterior component with $\ell_{\bar{w}}^{(i)}$ at time $k - 1$, and $\mathbf{U}_w(rn_{w_m}, 1)$ is the maximum weight of Gaussian components with $\ell_{\bar{w}}^{(i)}$.

② Survival target status information is obtained as follows:

$$\begin{aligned}\hat{\mathbf{X}}_k &= [\hat{\mathbf{X}}_k, \hat{\mathbf{x}}], \\ \mathcal{L}_{k,st} &= [\mathcal{L}_{k,st}, \ell_{\bar{w}}^{(i)}], \\ \mathcal{T}_{k,st} &= [\mathcal{T}_{k,st}, k].\end{aligned}\quad (29)$$

Additionally, $CN = [CN, cn^{(i)}]$.

③ Modifying the weights of the Gaussian components. The larger \bar{w} is, the more concentrated the posterior distribution is, and the smaller the radius of the effective region is. The smaller \bar{w} is, the more dispersed the posterior distribution is, and the larger the radius of the effective region is. By calculating the distances between the current $\hat{\mathbf{x}}$ and the components with the same label via (30), the weight is modified via (31).

$$d^{(i1,j)} = \|\hat{\mathbf{x}} - \mathbf{H}_k \mathbf{U}_m\left(RN_{\ell_{\bar{w}}^{(i)}}, j\right)\|, \forall i1 = 1, \dots, |RN_{\ell_{\bar{w}}^{(i)}}|, \forall j = 2, \dots, 1 + |Z_{k,ef}| \quad (30)$$

$$\mathbf{U}_w\left(RN_{\ell_{\bar{w}}^{(i)}}, j\right) = a_1 \bar{w}, \quad (\bar{w} \geq 0.5 \& d^{(i1,j)} > d(3)) \mid (0.4 \leq \bar{w} < 0.5 \& d^{(i1,j)} > d(4)) \mid (0.2 \leq \bar{w} < 0.4 \& d^{(i1,j)} > d(5)) \mid (\bar{w} < 0.2 \& d^{(i1,j)} > d(6)) \quad (31)$$

In the effective region, the weights of Gaussian components remain unchanged; however, they are reduced to $a_1 \bar{w}$ outside the region. Here, $a_1 = 0.3$. Dividing the effective region of the posterior distribution based on the weights can further shield against interference, such as detection uncertainty and closely spaced measurements originating from clutter or the other target.

- According to the “one-to-one” principle, the elements in $\mathbf{U}\left(RN_{\ell_{\bar{w}}^{(i)}}\right)$ are employed in the state extraction; then, they will be discarded as follows:

$$\mathbf{U}_w\left(RN_{\ell_{\bar{w}}^{(i)}}\right) = 0 \tag{32}$$

Step 3: Based on the components associated with effective observations, whose sequence numbers are stored in CN , some tracks are established or maintained. Thus, after the $i = 1, \dots, |\mathcal{L}_{\bar{w}}|$ loop is completed, the elements in $\mathbf{U}(:, CN)$ will be discarded as follows:

$$\mathbf{U}_w(:, CN) = 0 \tag{33}$$

Next, return to Step 1.

Step 4: Here, $nr_{\hat{\mathbf{U}}}$ denotes the number of rows in $\hat{\mathbf{U}}, \hat{\mathbf{U}}_w, \hat{\mathbf{U}}_m, \hat{\mathbf{U}}_p$, and $\hat{\mathbf{U}}_\ell$, which all have size $nr_{\hat{\mathbf{U}}} \times (1 + |Z_{k,ef}|)$. Converting these matrices into $1 \times (nr_{\hat{\mathbf{U}}}(1 + |Z_{k,ef}|))$ column by column, the parameter set of modified Gaussian components is obtained as follows:

$$\left\{ \tilde{w}_{k|k}^{(j)}, \tilde{m}_{k|k}^{(j)}, \tilde{P}_{k|k}^{(j)}, \tilde{\ell}_{k|k}^{(j)} \right\}_{j=1}^{nr_{\hat{\mathbf{U}}}(1+|Z_{k,ef}|)}$$

This completes the state extraction operation at time k .

The overall procedure of the state extraction of the LGM-PHD filter is given in Appendix A.

Scheme 1 illustrates the process for extracting multiple single-target state estimates based on \mathbf{U} . As the elements in each row in \mathbf{U}_ℓ have the same label, they are recorded at the left of each row in \mathbf{U}_w , as $\{401\}$, to simplify the space. In Scheme 1a, at time k , there are two classes of confirmed components: $\{1, 2\}$; one unconfirmed component: $\{203\}$; four classes of newborn components: $\{401, 402, 403, 404\}$; seven effective measurements selected via Section 3.2. At time $k - 1$, r_{\max} and $r_{\max t}$ are incremented to $r_{\max} = 2$ and $r_{\max t} = 203$. Initialize $\hat{X}_k = \emptyset, \mathcal{L}_{k,st} = \emptyset$, and $\mathcal{T}_{k,st} = \emptyset$.

	0	1	2	3	4	5	6	7
(401)	0.0003	0	0	0	0.5262	0.2616	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0.0003	0	0	0	0	0	0.4087	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0.0072	0.8256	0	0	0	0	0	0.4703
(1)	0.0001	0	0	0	0	0	0	0.0002
(2)	0.0002	0	0.2777	0	0	0	0	0
(203)	0.0025	0	0	0.5086	0	0	0	0

(a)

	0	1	2	3	4	5	6	7
(401)	0.0003	0	0	0	0.5262	0.2616	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0.0003	0	0	0	0	0	0.4087	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(2)	0.0002	0	0.2777	0	0	0	0	0
(203)	0.0025	0	0	0.5086	0	0	0	0

(b)

	0	1	2	3	4	5	6	7
(401)	0	0	0	0	0	0	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0.0003	0	0	0	0	0	0.4087	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(2)	0.0002	0	0.2777	0	0	0	0	0
(203)	0.0025	0	0	0.5086	0	0	0	0

(c)

	0	1	2	3	4	5	6	7
(401)	0	0	0	0	0	0	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0.0003	0	0	0	0	0	0.4087	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(2)	0.0002	0	0.2777	0	0	0	0	0
(203)	0	0	0	0	0	0	0	0

(d)

Scheme 1. Cont.

	0	1	2	3	4	5	6	7
(401)	0	0	0	0	0	0	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0	0	0	0	0	0	0	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(2)	0.0002	0	0.2777	0	0	0	0	0
(203)	0	0	0	0	0	0	0	0

(e)

	0	1	2	3	4	5	6	7
(401)	0	0	0	0	0	0	0	0
(402)	0.0003	0	0	0	0	0	0	0
(403)	0	0	0	0	0	0	0	0
(404)	0.0003	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(1)	0	0	0	0	0	0	0	0
(2)	0	0	0	0	0	0	0	0
(203)	0	0	0	0	0	0	0	0

(f)

Scheme 1. Steps of extracting multiple single-target state estimate based on \mathbf{U} . (a–f): the successive steps of extracting multiple single-target state estimates.

First, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.8256$, which is $\mathbf{U}_w(5, 2)$, as shown in Scheme 1a. As $\bar{w} > w_b$, obtain $\mathbf{U}_\ell(5, 2) = 1$ and initialize $\text{CN} = \emptyset$. Here, $|\mathcal{L}_{\bar{w}}| = 1$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(5, 2)$ and $\text{RN}_{\ell_{\bar{w}}^{(i)}} = [5, 6]$. As $\mathbf{U}_\ell(5, 2) < V_{un}$, it is the label of the confirmed component. Then, based on (29), survival target status information is obtained. $\text{CN} = [\text{CN}, 2]$. Via (32) and (33), $\mathbf{U}_w([5, 6], :) = 0$ and $\mathbf{U}_w(:, 2) = 0$, Scheme 1b is obtained. It is obvious that the interference of $\mathbf{U}_w(5, 8)$ caused by $\mathbf{z}_{k, \text{ef}}^{(7)}$ with state estimation is prevented, which represents the clutter close to $\mathbf{z}_{k, \text{ef}}^{(1)}$.

Second, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.5262$, which is $\mathbf{U}_w(1, 5)$, as shown in Scheme 1b. As $\bar{w} > w_b$, obtain $\mathbf{U}_\ell(1, 5) = 401$ and initialize $\text{CN} = \emptyset$. Here, $|\mathcal{L}_{\bar{w}}| = 1$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(1, 5)$ and $\text{RN}_{\ell_{\bar{w}}^{(i)}} = 1$. As $\mathbf{U}_\ell(1, 5) > V_{new}$, it is the label of the newborn component. Based on (20), $\text{CN}_{w_l} = 5$, and $\text{CN} = [\text{CN}, 5]$. Then, information on one new target status is established via (21), and r_{max} is incremented to 3. The information on this new confirmed track is established by (22).

Additionally, based on (23), $\text{CN}_{w_{bl}} = 6$, and $\text{CN} = [\text{CN}, 6]$. Then, one new unconfirmed track is established via (24–26). Here, $r_{\text{max}t}$ is incremented to 204. Via (32) and (33), $\mathbf{U}_w(1, :) = 0$ and $\mathbf{U}_w(:, [5, 6]) = 0$, Scheme 1c is obtained.

Third, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.5086$, which is $\mathbf{U}_w(8, 4)$, as shown in Scheme 1c. As $\bar{w} > w_b$, obtain $\mathbf{U}_\ell(8, 4) = 203$ and initialize $\text{CN} = \emptyset$. Here, $|\mathcal{L}_{\bar{w}}| = 1$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(8, 4)$ and $\text{RN}_{\ell_{\bar{w}}^{(i)}} = 8$. As $V_{un} < \mathbf{U}_\ell(8, 4) \leq V_{new}$, it is the label of the unconfirmed component. Since $\bar{w} > w_l$, a new target status information is obtained by (27) and $\text{CN} = [\text{CN}, 4]$. Via (32) and (33), $\mathbf{U}_w(8, :) = 0$ and $\mathbf{U}_w(:, 4) = 0$, Scheme 1d is obtained.

Fourth, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.4087$, which is $\mathbf{U}_w(3, 7)$, as shown in Scheme 1d. As $\bar{w} > w_b$, obtain $\mathbf{U}_\ell(3, 7) = 403$ and initialize $\text{CN} = \emptyset$. Here, $|\mathcal{L}_{\bar{w}}| = 1$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(3, 7)$ and $\text{RN}_{\ell_{\bar{w}}^{(i)}} = 3$. As $\mathbf{U}_\ell(3, 7) > V_{new}$, it is the label of the newborn component. Based on (20), $\text{CN}_{w_l} = 7$, and $\text{CN} = [\text{CN}, 7]$. Then, information on one new target status is established via (21), and r_{max} is incremented to 4. The information on this new confirmed track is established by (22). Via (32) and (33), $\mathbf{U}_w(3, :) = 0$ and $\mathbf{U}_w(:, 7) = 0$, Scheme 1e is obtained.

Fifth, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.2777$, which is $\mathbf{U}_w(7, 3)$, as shown in Scheme 1e. As $\bar{w} > w_b$, obtain $\mathbf{U}_\ell(7, 3) = 2$ and initialize $\text{CN} = \emptyset$. Here, $|\mathcal{L}_{\bar{w}}| = 1$, $\hat{\mathbf{x}} = \mathbf{H}_k \mathbf{U}_m(7, 3)$ and $\text{RN}_{\ell_{\bar{w}}^{(i)}} = 7$. As $\mathbf{U}_\ell(7, 3) < V_{un}$, it is the label of the confirmed component. Then, based on (29), survival target status information is obtained. $\text{CN} = [\text{CN}, 3]$. Via (32) and (33), $\mathbf{U}_w(7, :) = 0$ and $\mathbf{U}_w(:, 3) = 0$, Scheme 1f is obtained.

Sixth, the maximum weight in $\mathbf{U}_w(:, 2 : \text{end})$ is $\bar{w} = 0.0003$ in Scheme 1f. As $\bar{w} < w_b$, step 1 to step 3 of the state extraction at time k is completed. $\hat{\mathbf{X}}_k$, $\mathcal{L}_{k, \text{st}}$, and $\mathcal{T}_{k, \text{st}}$ are all updated.

Remark 2. In the proposed filter, since the components approximating one survival target can be identified, the weight threshold for state extraction is reduced compared with the threshold in the basic GM-PHD filter. This ensures that the survival target can be tracked when the proper measurement error occurs, such as $\mathbf{U}_w(7,3) = 0.2777$ in Scheme 1. Additionally, only one state estimate can be extracted from the survival Gaussian components with the same label. This ensures that the interference of clutter on the number of state estimates, such as $\mathbf{U}_w(5,8) = 0.4703$ in Scheme 1, can be prevented.

Remark 3. The behavior of labeling state estimates in the proposed filter is determined by employing the weights and labels of the updated components, which requires no additional computational processing. Furthermore, the updating process and query tables in the state extraction algorithm are all based on the selected observations. Thus, the real-time performance of the LGM-PHD filter is guaranteed.

3.5. Pruning

To reduce the computational cost, reducing the number of Gaussian components by pruning is necessary.

Step 1: Prune confirmed and unconfirmed components that have no state estimates for successive n_{no} time steps. In our filter, $n_{no} = 6$.

① The label set $\mathcal{L}_{k,no}$ of confirmed and unconfirmed components with no state estimates at time k is:

$$\mathcal{L}_{k,no} = \left\{ \ell_{k,no}^{(i)} \right\}_{i=1}^{|\mathcal{L}_{k,no}|} = \text{uni} \left(\left\{ \tilde{\ell}_{k|k}^{(j)} \right\}_{j=1}^{nr \hat{\mathbf{u}}^{(1+|Z_{k,ef}|)}} \right) - \left\{ \ell_{\gamma,k}^{(i)} \right\}_{i=1}^{J_{\gamma,k}} - \mathcal{L}_{k,st} \quad (34)$$

② For $i = 1, \dots, |\mathcal{L}_{k,no}|$, check the occurrence number of $\ell_{k,no}^{(i)}$ in $\{\mathcal{L}_{k1,no}\}_{k1=k-n_{no}+1}^{k-1}$. If $\ell_{k,no}^{(i)}$ appears in $\mathcal{L}_{k1,no}$ at each time $k1 = k - n_{no} + 1, \dots, k - 1$, $\left\{ \tilde{w}_{k|k}^{(j)} = 0 \mid \tilde{\ell}_{k|k}^{(j)} = \ell_{k,no}^{(i)} \right\}_{j=1}^{nr \hat{\mathbf{u}}^{(1+|Z_{k,ef}|)}}$. This procedure improves accuracy.

Step 2: Based on the truncation threshold $w_{et} = 10^{-5}$, a certain number of the components with the strongest weights are retained in I ,

$$I = \left\{ j = 1, \dots, nr \hat{\mathbf{u}}^{(1+|Z_{k,ef}|)} \mid \tilde{w}_{k|k}^{(j)} > w_{et} \right\} \quad (35)$$

Step 3: Obtain $\left\{ \tilde{w}_{k|k}^{(j)}, \tilde{\mathbf{m}}_{k|k}^{(j)}, \tilde{\mathbf{P}}_{k|k}^{(j)}, \tilde{\ell}_{k|k}^{(j)} \right\}_{j=1}^{J_{k|k}}$, where $J_{k|k} = |I|$, $(\tilde{w}_{k|k}^{(j)}, \tilde{\mathbf{m}}_{k|k}^{(j)}, \tilde{\mathbf{P}}_{k|k}^{(j)}, \tilde{\ell}_{k|k}^{(j)} \mid j = 1) = (\tilde{w}_{k|k}^{(j)}, \tilde{\mathbf{m}}_{k|k}^{(j)}, \tilde{\mathbf{P}}_{k|k}^{(j)}, \tilde{\ell}_{k|k}^{(j)} \mid j = I(1))$, among others.

3.6. Merging and Capping

In the basic GM-PHD filter, some of the Gaussian components that are very close together will be approximated by a single Gaussian [32]. This means that, in the merging procedure, the posterior information of targets in close proximity will interfere. This will result in the loss of posterior information, approximated by components with small weights, of some targets. To avoid such loss, in the proposed filter, only the components with the same label can be merged.

First, let d_{Th} be the merging threshold. In general, $d_{Th} = 4$. Set $J'_k = 0$, and $I1 = \{i1\}_{i1=1}^{J_{k|k}}$. The specific steps in merging process are as follows.

Step 1: Let $J'_k = J'_k + 1$, $j = \arg \max_{i1 \in I1} (\tilde{w}_{k|k}^{(i1)})$. Then, $\tilde{\ell}_{k|k}^{(j)}$ is obtained.

Step 2: The sequence number set $I2$ is given as follows:

$$I2 = \left\{ i1 \in I1 \left| \widehat{\ell}_{k|k}^{(i1)} = \widehat{\ell}_{k|k}^{(j)}, (\widehat{\mathbf{m}}_{k|k}^{(i1)} - \widehat{\mathbf{m}}_{k|k}^{(j)})^T \left(\widehat{\mathbf{P}}_{k|k}^{(i1)} \right)^{-1} (\widehat{\mathbf{m}}_{k|k}^{(i1)} - \widehat{\mathbf{m}}_{k|k}^{(j)}) \leq d_{Th} \right. \right\} \quad (36)$$

Step 3: Then, a merged Gaussian component is given as follows:

$$\widetilde{w}_k^{(J'_k)} = \sum_{i1 \in I2} \widehat{w}_{k|k}^{(i1)} \quad (37)$$

$$\widetilde{\mathbf{m}}_k^{(J'_k)} = \frac{1}{\widetilde{w}_k^{(J'_k)}} \sum_{i1 \in I2} \widehat{w}_{k|k}^{(i1)} \widehat{\mathbf{m}}_{k|k}^{(i1)} \quad (38)$$

$$\widetilde{\mathbf{P}}_k^{(J'_k)} = \frac{1}{\widetilde{w}_k^{(J'_k)}} \sum_{i1 \in I2} \widehat{w}_{k|k}^{(i1)} \left(\widehat{\mathbf{P}}_{k|k}^{(i1)} + \left(\widehat{\mathbf{m}}_{k|k}^{(i1)} - \widetilde{\mathbf{m}}_k^{(J'_k)} \right) \left(\widehat{\mathbf{m}}_{k|k}^{(i1)} - \widetilde{\mathbf{m}}_k^{(J'_k)} \right)^T \right) \quad (39)$$

$$\widetilde{\ell}_k^{(J'_k)} = \widehat{\ell}_{k|k}^{(j)} \quad (40)$$

Step 4: Execute $I1 = I1 - I2$. If $I1$ is not empty, skip to Step 1.

Step 5: The output of merging procedure is obtained as follows: $\left\{ \widetilde{w}_k^{(j)}, \widetilde{\mathbf{m}}_k^{(j)}, \widetilde{\mathbf{P}}_k^{(j)}, \widetilde{\ell}_k^{(j)} \right\}_{j=1}^{J'_k}$.

Next, if $J'_k > J_{\max 1}$, replace $\left\{ \widetilde{w}_k^{(j)}, \widetilde{\mathbf{m}}_k^{(j)}, \widetilde{\mathbf{P}}_k^{(j)}, \widetilde{\ell}_k^{(j)} \right\}_{j=1}^{J'_k}$ by those of the $J_{\max 1}$ Gaussian components with the largest weights, and the posterior intensity for propagating is obtained as follows: $\left\{ w_k^{(j)}, \mathbf{m}_k^{(j)}, \mathbf{P}_k^{(j)}, \ell_k^{(j)} \right\}_{j=1}^{J_k}$. In this paper, $J_{\max 1} = 150$.

3.7. Steps in the Proposed Algorithm

For completeness, we summarized the key steps of the LGM-PHD filter as follows.

Step 1: Prediction. The predicted components $\left\{ w_{k|k-1}^{(i)}, \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)} \right\}_{i=1}^{J_{k|k-1}}$ are obtained via (4) and their labels are obtained via (16). Then, obtain the parameter set $\left\{ w_{k|k-1}^{(i)}, \mathbf{m}_{k|k-1}^{(i)}, \mathbf{P}_{k|k-1}^{(i)}, \ell_{k|k-1}^{(i)} \right\}_{i=1}^{J_{k|k-1}}$ of the predicted components.

Step 2: Observation selection. The effective observation set $Z_{k,ef}$ is obtained by eliminating the clutter in the low prior region via Algorithm 1.

Step 3: Updating. Based on $Z_{k,ef}$, the predicted components are updated via (5) to (12), and the parameters $w, \mathbf{m}, \mathbf{P}$, and ℓ associated with each component are stored in \mathbf{U} and $\widehat{\mathbf{U}}$, respectively.

Step 4: State extraction. As per the steps in 3.4, based on \mathbf{U} and $\widehat{\mathbf{U}}$, the multitarget state extraction is converted into multiple single-target state extractions that can provide the identity label, and the occurrence time for each estimate, i.e., $\widehat{X}_k, \mathcal{L}_{k,st}$, and $\widehat{T}_{k,st}$ are obtained. Then, converting the four $\widehat{\mathbf{U}}$ values into $1 \times (nr_{\widehat{\mathbf{U}}} (1 + |Z_{k,ef}|))$ column by column, the parameter set $\left\{ \widetilde{w}_{k|k}^{(j)}, \widetilde{\mathbf{m}}_{k|k}^{(j)}, \widetilde{\mathbf{P}}_{k|k}^{(j)}, \widetilde{\ell}_{k|k}^{(j)} \right\}_{j=1}^{nr_{\widehat{\mathbf{U}}} (1 + |Z_{k,ef}|)}$ of the modified Gaussian components is obtained.

Step 5: Pruning. As per the steps in Section 3.5, $\left\{ \widetilde{w}_{k|k}^{(j)}, \widetilde{\mathbf{m}}_{k|k}^{(j)}, \widetilde{\mathbf{P}}_{k|k}^{(j)}, \widetilde{\ell}_{k|k}^{(j)} \right\}_{j=1}^{J_{k|k}}$ are obtained.

Step 6: Merging and capping. The closely spaced components with the same label are merged as per the steps in Section 3.6 to obtain $\left\{ w_k^{(j)}, \mathbf{m}_k^{(j)}, \mathbf{P}_k^{(j)}, \ell_k^{(j)} \right\}_{j=1}^{J_k}$ for propagation.

After Step 4, based on \hat{X}_k , $\mathcal{L}_{k,st}$, and $\mathcal{T}_{k,st}$, explicit track maintenance is achieved by connecting the state estimates with the same label in sequence.

Remark 4. The proposed filter does not specifically define track disappearance, and each track is left as it is, whether it has disappeared or been misdetections, except that no state estimate appears in seven successive time steps. When a target has been redetected, as long as confirmed Gaussian components for this target are still available for propagation, this target can still have clues to be tracked, and the state estimate will be extracted from these components only if its posterior information satisfies the necessary conditions.

4. Simulation and Results Analysis

The experiment presented in this section aimed to test the performance of the proposed filter. We compared the proposed filter with the basic GM-PHD filter [32] and the Gibbs-GLMB filter [10]. Without loss of generality, we employed the MTT simulation model given in [39]. In a two-dimensional scenario over the region $[-1000 \ 1000] \text{ m} \times [-1000 \ 1000] \text{ m}$, each target moved according to the following linear Gaussian dynamics:

$$\mathbf{x}_k = \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \sigma_\omega^2 \begin{bmatrix} \Delta^2/4 & \Delta^3/2 & 0 & 0 \\ \Delta^3/2 & \Delta^2 & 0 & 0 \\ 0 & 0 & \Delta^2/4 & \Delta^3/2 \\ 0 & 0 & \Delta^3/2 & \Delta^2 \end{bmatrix}$$

where $\mathbf{x}_k = [x_{1,k} \ x_{2,k} \ x_{3,k} \ x_{4,k}]^T$, $[x_{1,k} \ x_{3,k}]^T$, and $[x_{2,k} \ x_{4,k}]^T$ are, respectively, the position and velocity of the target at time k . $\Delta = 1 \text{ s}$ is the sampling time, and $\sigma_\omega = 5 \text{ m/s}^2$. Targets can appear and disappear in the scene at any time with the survival probability $p_{S,k} = 0.95$. For the Gibbs-GLMB filter, the birth model is a labeled multi-Bernoulli with parameters $\{r_{B,k}(\ell_i), p_{B,k}(\ell_i)\}_{i=1}^4$, where $\ell_i = (k, i)$, $r_{B,k}(\ell_i) = 0.03$ and $p_{B,k}(\ell_i) = \mathcal{N}(\mathbf{x}; \mathbf{m}_B^{(i)}, \mathbf{P}_B)$ with $\mathbf{m}_B^{(1)} = [0 \ 0 \ 0 \ 0]^T$, $\mathbf{m}_B^{(2)} = [400 \ 0 \ -600 \ 0]^T$, $\mathbf{m}_B^{(3)} = [-800 \ 0 \ -200 \ 0]^T$, $\mathbf{m}_B^{(4)} = [-200 \ 0 \ 800 \ 0]^T$, and $\mathbf{P}_B = \text{diag}([10 \ 10 \ 10 \ 10]^T)^2$. For the GM-PHD filter, the newborn targets appear spontaneously according to a Poisson point process with intensity function $\gamma_k(\mathbf{x}_k) = \sum_{i=1}^4 w_{\gamma,k}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^{(i)}, \mathbf{P}_{\gamma,k}^{(i)})$, which are given by $\mathbf{m}_{\gamma,k}^{(1)} = \mathbf{m}_B^{(1)}$, $\mathbf{m}_{\gamma,k}^{(2)} = \mathbf{m}_B^{(2)}$, $\mathbf{m}_{\gamma,k}^{(3)} = \mathbf{m}_B^{(3)}$, $\mathbf{m}_{\gamma,k}^{(4)} = \mathbf{m}_B^{(4)}$, $\mathbf{P}_{\gamma,k} = \mathbf{P}_B$, and $w_{\gamma,k}^{(i)} = r_{B,k}(\ell_i)$. Additionally, $\ell_{\gamma,k}^{(1)} = 401$, $\ell_{\gamma,k}^{(2)} = 402$, $\ell_{\gamma,k}^{(3)} = 403$, and $\ell_{\gamma,k}^{(4)} = 404$.

The linear target-originated measurement is given as follows:

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} v_{x,k} \\ v_{y,k} \end{bmatrix}$$

where $v_{x,k}$ and $v_{y,k}$ represent mutually independent zero-mean Gaussian white noise with standard deviations of $\sigma_x = 10 \text{ m}$ and $\sigma_y = 10 \text{ m}$, respectively. The clutter is uniformly distributed over the region $[-1000 \ 1000] \text{ m} \times [-1000 \ 1000] \text{ m}$ with an average rate of λ points per scan, i.e., $\kappa = \lambda/2000^2$, for the basic GM-PHD filter. To ensure a fair comparison, the basic GM-PHD filter employs the gating algorithm that is used in Gibbs-GLMB filter, and $G_{th} = 9$.

The maximum number of Gaussian components is $J_{\max} = 100$ in the proposed filter and the basic GM-PHD filter. The optimal subpattern assignment (OSPA) metric [40] is used to evaluate the tracking performance of different filters, and the order and cut-off are set as $p = 1$ and $c = 100$, respectively. The presented result is the average of 250 independent repeated experiments but with independently generated observations for each trial. All

experiments were run in MATLAB R2013b on a computer with a 3.3 GHz CPU and 8 GB RAM.

Example 1: Detailed performance comparison under $\lambda = 100$ and $p_{D,k} = 0.9$. As shown in Figures 4–6, the number of clutter points is $\lambda = 100$, and the detection probability is $p_{D,k} = 0.9$. Figure 4 shows the track estimates obtained from the proposed filter for a single trial [38]. The average OSPA errors and computing time for 250 sample runs of each filter are shown in Figures 5 and 6.

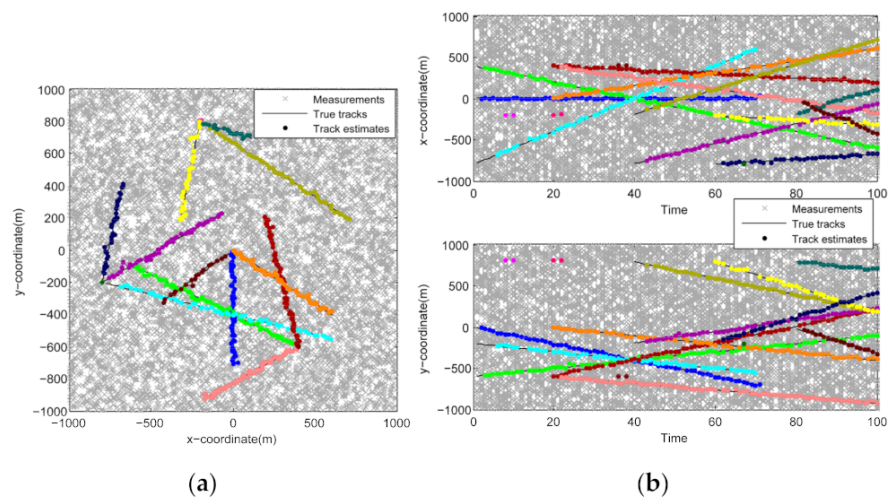


Figure 4. Track estimates of the proposed filter and ground truths for a sample run ($\lambda = 100$, $p_{D,k} = 0.9$). (a) The true and estimated trajectories of targets. Different trajectories estimated by the proposed filter are denoted by different colored dots. (b) The true and estimated trajectories of targets in x and y coordinates, respectively. © 2021 IEEE.

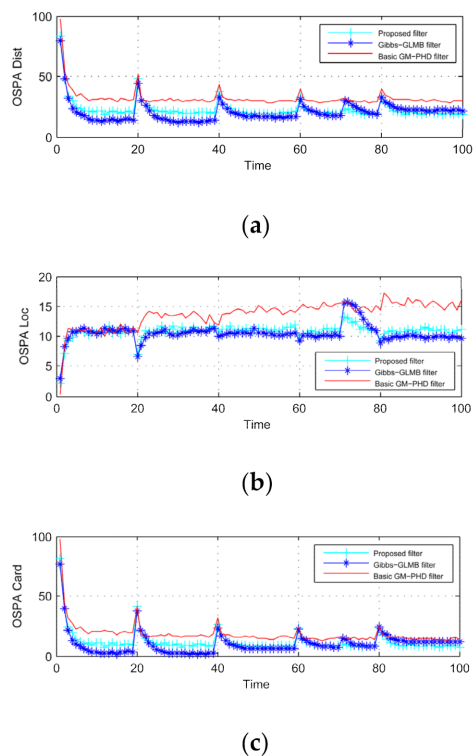


Figure 5. Comparison of different algorithms for 250 sample runs ($\lambda = 100$, $p_{D,k} = 0.9$). (a) Average Optimal subpattern assignment (OSPA) distances of different filters; (b) OSPA locations of different filters; (c) OSPA cardinalities of different filters.

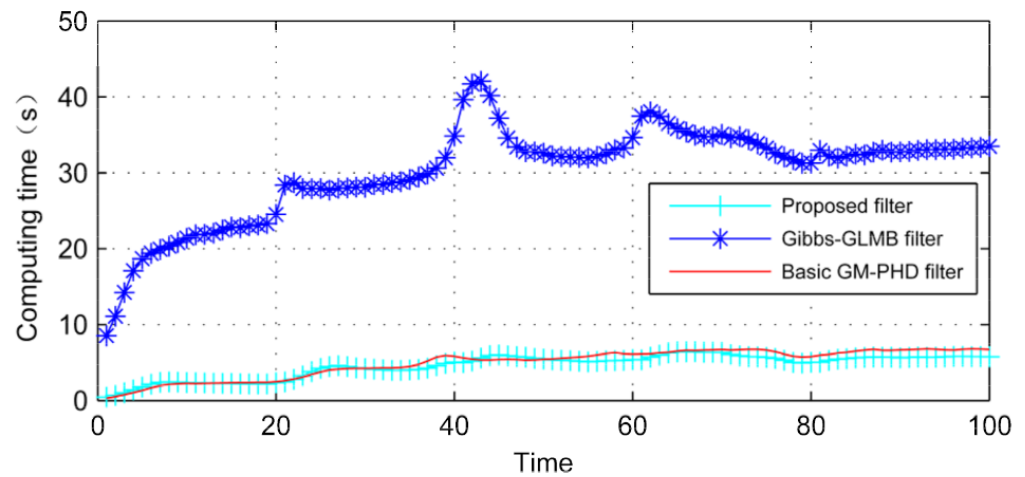


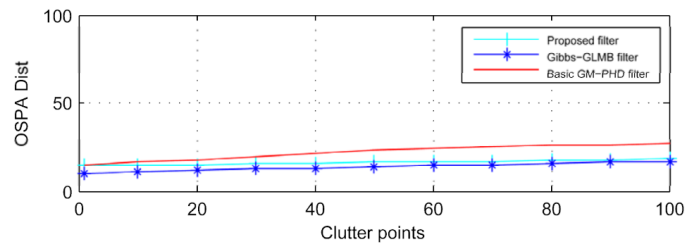
Figure 6. Computing times of different algorithms for 250 sample runs ($\lambda = 100, p_{D,k} = 0.9$).

Figure 4a shows that the proposed filter can achieve correct trajectory maintenance in MTT in the presence of dense clutter. As the proposed filter does not supplement the state estimate for the misdetected target, there are discontinuities in the tracks. However, this does not affect the trajectory continuity, and the redetected targets can be tracked in time. Additionally, as shown in Figure 4b, one false track is around $H_k m_{\gamma,k}^{(2)}$, one false track is around $H_k m_{\gamma,k}^{(3)}$ and two false tracks are around $H_k m_{\gamma,k}^{(4)}$. This is because the proposed filter has no special procedure to account for the false estimates around the newborn locations. Similarly, they do not affect the continuity of other established tracks, and will disappear over time.

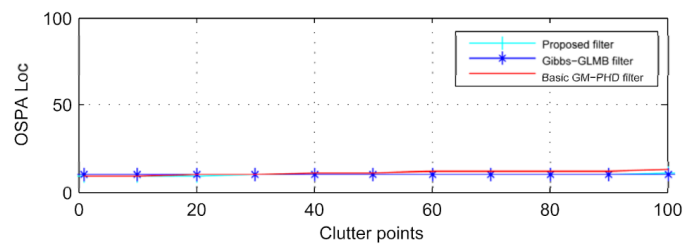
In Figure 5, the average OSPA distances of the proposed filter, the Gibbs-GLMB filter and the basic GM-PHD filter were 22.522 m, 20.497 m, and 31.990 m, respectively. Compared with the basic GM-PHD filter, the average tracking performance of the proposed filter and the Gibbs-GLMB filter increased by 29.60% and 35.92%, respectively. Clearly, the tracking performance of the LGM-PHD filter was slightly lower than that of the Gibbs-GLMB filter. This is mainly caused by the phenomena shown in Figure 4b. However, this did not affect the survival target tracking.

As shown in Figure 6, the average computing times in seconds of a single run (100 time steps) of our MATLAB implementations were 4.60 s (the proposed filter), 30.21 s (the Gibbs-GLMB filter), and 4.96 s (the basic GM-PHD filter), respectively. Compared with the basic GM-PHD filter, the real-time performance of the proposed filter and the Gibbs-GLMB filter increased by 7.26% and -509% , respectively. As expected, the real-time performance of the proposed filter was better than that of the Gibbs-GLMB filter.

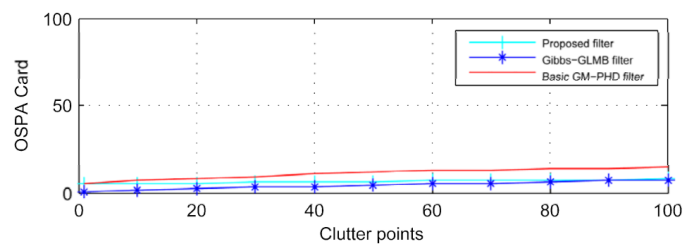
Example 2: Performance comparison under various clutter rates for $p_{D,k} = 0.95$. In this example, the performance of the proposed filter was evaluated with various clutter rates. The clutter rate λ changed from 1 to 100, and the probability of detection was set to $p_{D,k} = 0.95$. Figure 7 shows the comparison results in terms of the average OSPA errors for 250 sample runs of the different algorithms under different clutter rates, and their computing times are shown in Figure 8.



(a)



(b)



(c)

Figure 7. Comparison of different algorithms under various clutter rates for 250 sample runs ($p_{D,k} = 0.95$). (a) Average OSPA distances of different filters; (b) OSPA locations of different filters; (c) OSPA cardinalities of different filters.

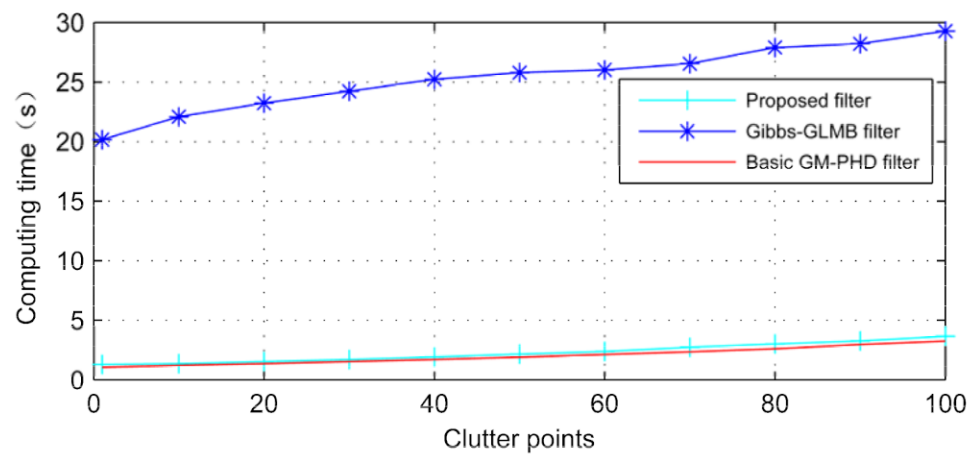
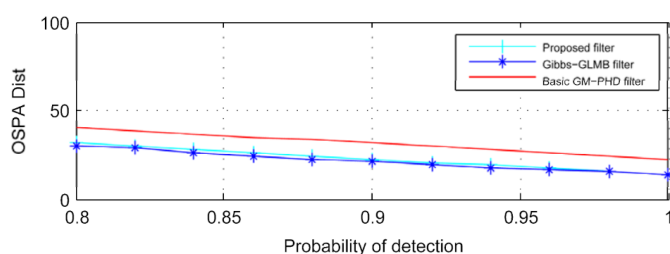


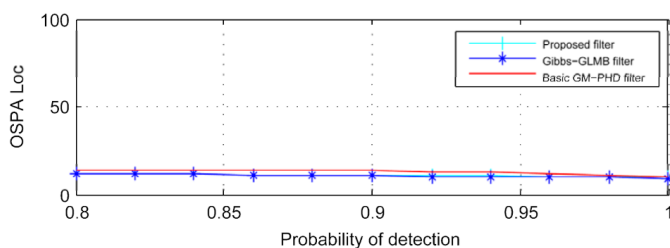
Figure 8. Computing times of different algorithms under various clutter rates for 250 sample runs ($p_{D,k} = 0.95$).

From Figures 7 and 8, we can see that the proposed filter achieved a better OSPA distance than the basic GM-PHD filter and kept a similar real-time performance as the clutter rate increased. Compared with the basic GM-PHD filter, with $\lambda = 20$, the average tracking performances of the proposed filter and the Gibbs-GLMB filter increased by 15.75% and 33.14%, respectively; with $\lambda = 100$, they increased by 31.96% and 37.14%, respectively. The advantage of the proposed filter can be fully demonstrated in the dense clutter environment.

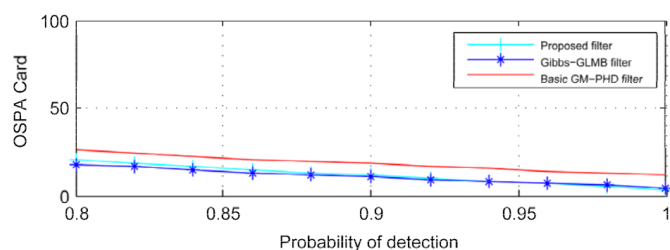
Example 3: Performance comparison under various probabilities of detection for $\lambda = 100$. In this example, various probabilities of detection were employed to evaluate the effectiveness of the proposed filter. The probability of detection $p_{D,k}$ changed from 0.8 to 1, and the clutter rate was kept unchanged at $\lambda = 100$. The comparison results of the average OSPA errors for 250 sample runs of the different algorithms under different probabilities of detection are shown in Figure 9. Figure 10 shows the corresponding computing times of each filter.



(a)



(b)



(c)

Figure 9. Comparison of different algorithms under various probabilities of detection for 250 sample runs ($\lambda = 100$). (a) Average OSPA distances of different filters; (b) OSPA locations of different filters; (c) OSPA cardinalities of different filters.

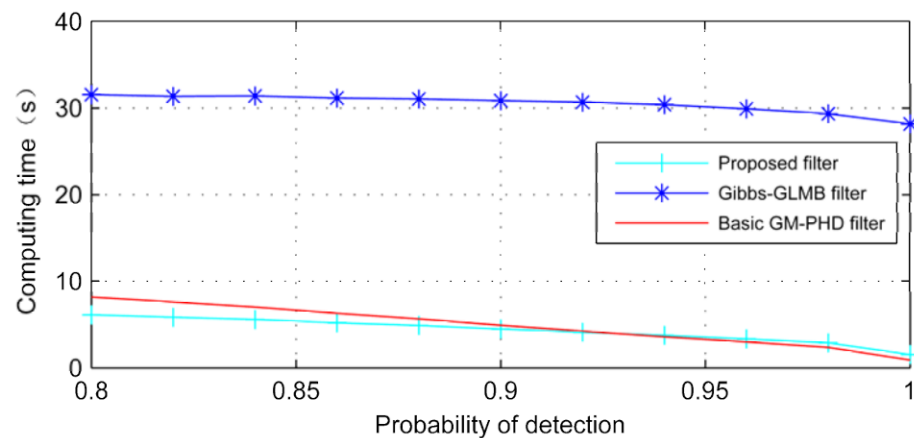


Figure 10. Computing times of different algorithms under various probabilities of detection for 250 sample runs ($\lambda = 100$).

Clearly, the proposed filter achieved a better OSPA distance than the basic GM-PHD filter and maintained a similar real-time performance at each probability of detection. As the probability of detection increased, the OSPA distance of the proposed filter decreased. Compared with the basic GM-PHD filter, with $p_{D,k} = 0.84$, the average tracking performances of the proposed filter and the Gibbs-GLMB filter increased by 23.03% and 29.13%, respectively; with $p_{D,k} = 0.96$, they increased by 33.21% and 35.79%, respectively.

5. Conclusions

The paper proposed a computationally efficient LGM-PHD filter that can explicitly accommodate the estimation of target trajectories in MTT. Based on the proposed query matrices, state estimates with identity labels and occurrence times were extracted, and no additional associated procedures were required. Three numerical and simulations analyses demonstrated that the LGM-PHD filter could effectively improve the tracking performance as compared with the basic GM-PHD filter, which was slightly lower than that of the Gibbs-GLMB filter. As the proposed filter could maintain a similar to or slightly lower real-time performance than that of the basic GM-PHD filter, it could be an alternative filter explicitly accommodating target trajectories in real-time processing applications. We should point out that the LGM-PHD filter is not suitable for low detection scenarios. In our future work, we will investigate multisensor technology [12,13,39] for improving tracking performance.

Author Contributions: Y.G. performed the simulations and wrote the paper; D.J. and C.Z. revised this paper and offered some useful suggestions in methodologies and simulations; S.G. offered revision during the whole process. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Fundamental Research Funds for the Central Universities under grant B200202165 and by National Natural Science Foundation of China under grant 61971179.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

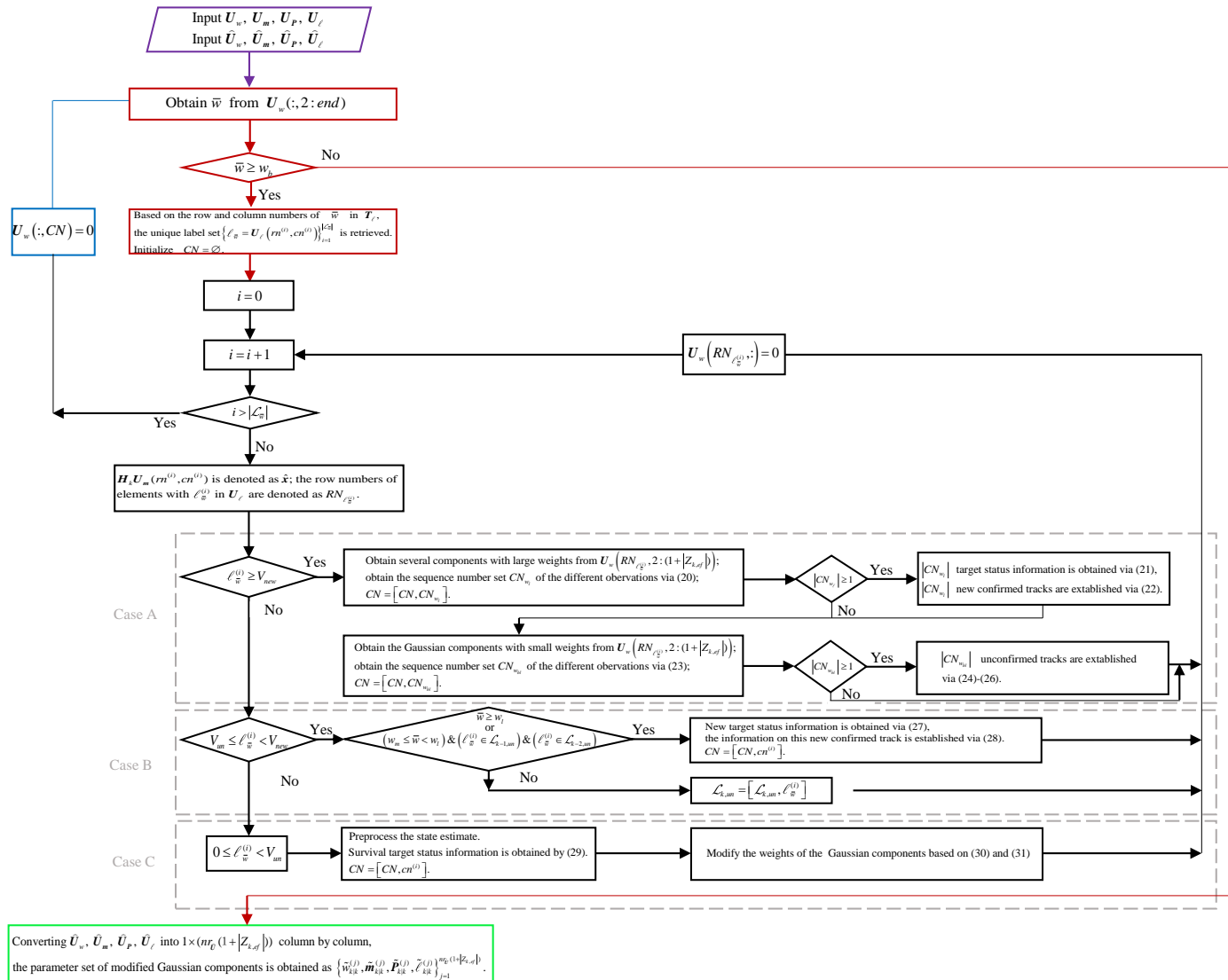


Figure A1. The overall procedure of the state extraction. The operations in the red boxes are step 1, the operations in the black boxes are step 2, the operation in the blue box is step 3, and the operations in the green box are step 4.

References

1. Mahler, R.P.S. *Statistical Multisource-Multitarget Information Fusion*; Artech House: Norwood, MA, USA, 2007.
2. Blackman, S.; Popoli, R. *Design and Analysis of Modern Tracking Systems*; Artech House: Norwood, MA, USA, 1999.
3. Bar-Shalom, Y.; Willett, P.K.; Tian, X. *Tracking and Data Fusion: A Handbook of Algorithms*; YBS Publishing: Storrs, CT, USA, 2011.
4. Blackman, S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerosp. Electron. Syst. Mag.* **2004**, *19*, 5–18. [[CrossRef](#)]
5. Jiang, D.; Li, M.; Gao, Y.; Gao, Y.; Fu, W.; Han, Y. Time-Matching Random Finite Set-Based Filter for Radar Multi-Target Tracking. *Sensors* **2018**, *18*, 4416. [[CrossRef](#)]
6. Mahler, R.P.S. Multi-target Bayes filtering via first-order multi-target moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [[CrossRef](#)]
7. Gao, Y.; Jiang, D.; Liu, M. Particle-gating SMC-PHD filter. *Signal Process.* **2017**, *130*, 64–73. [[CrossRef](#)]
8. Mahler, R.P.S. PHD filters of higher order in target number. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 1523–1543. [[CrossRef](#)]
9. Vo, B.T.; Vo, B.N.; Cantoni, A. The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE Trans. Signal Process.* **2009**, *57*, 409–423.
10. Vo, B.N.; Vo, B.T.; Hoang, H. An Efficient Implementation of the Generalized Labeled Multi-Bernoulli Filter. *IEEE Trans. Signal Process.* **2017**, *65*, 1975–1987. [[CrossRef](#)]
11. Wang, S.; Bao, Q.; Chen, Z. Refined PHD Filter for Multi-target Tracking under Low Detection Probability. *Sensors* **2019**, *19*, 2842. [[CrossRef](#)]
12. Chai, L.; Yi, W.; Jiang, X.; Kong, L. A Distributed PHD Filter for On-line Joint Sensor Registration and Multi-target Tracking. In Proceedings of the International Conference on Information Fusion, Ottawa, ON, Canada, 2–5 July 2019; pp. 1–6.
13. Li, T.; Mallick, M.; Pan, Q. A parallel Filtering-Communication-Based Cardinality Consensus Approach for Real-Time Distributed PHD Filtering. *IEEE Sens. J.* **2020**, *20*, 13824–13832. [[CrossRef](#)]
14. Vo, B.T.; Vo, B.N. Labeled Random Finite Sets and Multi-Object Conjugate Priors. *IEEE Trans. Signal Process.* **2013**, *61*, 3460–3475. [[CrossRef](#)]
15. Reuter, S.; Vo, B.T.; Vo, B.N.; Dietmayer, K. The labelled multi-Bernoulli filter. *IEEE Trans. Signal Process.* **2014**, *62*, 3246–3260.
16. Tian, Z.; Liu, W.; Ru, X. Multi-Target Localization and Tracking Using TDOA and AOA Measurement Based on Gibbs-GLMB Filtering. *Sensors* **2019**, *19*, 5437. [[CrossRef](#)]
17. Liu, R.; Fan, H.; Li, T.; Xiao, H. A Computationally Efficient Labeled Multi-Bernoulli Smoother for the Multi-Target Tracking. *Sensors* **2019**, *19*, 4226. [[CrossRef](#)]
18. García-Fernández, Á.F.; Svensson, L.; Williams, J.L.; Xia, Y.; Granström, K. Trajectory Poisson Multi-Bernoulli Filters. *IEEE Trans. Signal Process.* **2020**, *68*, 4933–4945. [[CrossRef](#)]
19. Liang, G.; Li, Q.; Qi, B.; Qiu, L. Multitarget Tracking Using One Time Step Lagged Delta-Generalized Labeled Multi-Bernoulli Smoothing. *IEEE Access.* **2020**, *8*, 28242–28256. [[CrossRef](#)]
20. Reuter, S.; Danzer, A.; Stübler, M.; Scheel, A.; Granström, K. A fast implementation of the Labeled Multi-Bernoulli filter using gibbs sampling. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 765–772.
21. Beard, M.; Reuter, S.; Granström, K.; Vo, B.T.; Vo, B.N.; Scheel, A. Multiple Extended Target Tracking With Labeled Random Finite Sets. *IEEE Trans. Signal Process.* **2016**, *64*, 1638–1653. [[CrossRef](#)]
22. García-Fernández, Á.F.; Svensson, L. Trajectory probability hypothesis density filter. In Proceedings of the International Conference on Information Fusion, Cambridge, UK, 10–13 July 2018; pp. 1430–1437.
23. Granström, K.; Willett, P.; Bar-Shalom, Y. Approximate Multi-Hypothesis Multi-Bernoulli Multi-Object Filtering Made Multi-Easy. *IEEE Trans. Signal Process.* **2016**, *64*, 1784–1797. [[CrossRef](#)]
24. Lin, L.; Bar-shalom, Y.; Kirubarajan, T. Tracking labeling and PHD filter for multitarget tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2006**, *42*, 778–795. [[CrossRef](#)]
25. Yang, J.; Ji, H. A novel track maintenance algorithm for PHD/CPHD filter. *Signal Process.* **2012**, *92*, 2371–2380. [[CrossRef](#)]
26. Li, Y.; Xiao, H.; Wu, H.; Fu, Q.; Hu, R. Modified Labeled Particle Probability Hypothesis Density Filter for Joint Multi-target Tracking and Classification. In Proceedings of the International Conference on Signal Processing and Communication Systems, Cairns, QLD, Australia, 14–16 December 2015; pp. 1–5.
27. Li, T.; Sun, S.; Corchado, J.M.; Siyan, M.F. A particle dyeing approach for track continuity for the SMC-PHD filter. In Proceedings of the International Conference on Information Fusion, Salamanca, Spain, 7–10 July 2014; pp. 1–8.
28. Cao, C.; Zhao, Y.; Pang, X.; Suo, Z.; Chen, S. An efficient implementation of multiple weak targets tracking filter with labeled random finite sets for marine radar. *Digital Signal Process.* **2020**, *101*, 102710. [[CrossRef](#)]
29. Gao, Y.; Jiang, D.; Liu, M.; Fu, W. An Explicit Track Continuity Algorithm for the SMC-PHD filter. In Proceedings of the International Conference on Communication Technology, Chengdu, China, 27–30 October 2017; pp. 1702–1708.
30. García-Fernández, Á.F.; Svensson, L. Trajectory PHD and CPHD Filters. *IEEE Trans. Signal Process.* **2019**, *67*, 5702–5714. [[CrossRef](#)]
31. Vo, B.N.; Singh, S.; Doucet, A. Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1224–1245.

32. Vo, B.N.; Ma, W.K. The Gaussian mixture probability hypothesis density filter. *IEEE Trans. Signal Process.* **2006**, *54*, 4091–4104. [[CrossRef](#)]
33. Zhang, H.; Ge, H.; Yang, J.; Yuan, Y. A GM-PHD algorithm for multiple target tracking based on false detection irregular window. *Signal Process.* **2016**, *120*, 537–552. [[CrossRef](#)]
34. Choi, M.E.; Seo, S.W. Robust Multitarget Tracking Scheme Based on Gaussian Mixture Probability Hypothesis Density Filter. *IEEE Trans. Veh. Technol.* **2016**, *65*, 4217–4229. [[CrossRef](#)]
35. Dehkordi, M.; Azimifar, Z. Refined GM-PHD tracker for tracking targets in possible subsequent missed detections. *Signal Process.* **2015**, *116*, 112–126. [[CrossRef](#)]
36. Gao, L.; Liu, H.; Liu, H. Probability hypothesis density filter with imperfect detection probability for mutli-target tracking. *Optik* **2016**, *127*, 10428–10436. [[CrossRef](#)]
37. Zhang, H.; Gao, L.; Xu, M.; Wang, Y. An improved probability hypothesis density filter for multi-target tracking. *Optik* **2019**, *182*, 23–31. [[CrossRef](#)]
38. Gao, Y.; Fang, N.; Jiang, D.; Fu, W.; Guo, S. An explicit track continuity algorithm for the GM-PHD filter. In Proceedings of the 31st Chinese Control and Decision Conference, Nanchang, China, 3–5 June 2019; pp. 1072–1079.
39. Vo, B.N.; Vo, B.T. An implementation of the Multi-sensor Generalized Labeled Multi-Bernoulli Filter via Gibbs Sampling. In Proceedings of the International Conference on Information Fusion, Xi'an, China, 10–13 July 2017; pp. 1–8.
40. Schuhmacher, D.; Vo, B.T.; Vo, B.N. A consistent metric for performance evaluation in multi-object filters. *IEEE Trans. Signal Process.* **2008**, *56*, 3447–3457. [[CrossRef](#)]