Label GM-PHD Filter Based on Threshold Separation Clustering

Kuiwu Wang 1,2,*, Qin Zhang 1 and Xiaolong Hu 1

1 School of Air Defense and Missile Defense, Air Force Engineering University, Xi’an 710051, China; kinzh@263.net (Q.Z.); huxiaolong@126.com (X.H.)
2 Graduate School, Air Force Engineering University, Xi’an 710051, China
* Correspondence: wkw19971997@163.com

Abstract: Gaussian mixture probability hypothesis density (GM-PHD) filtering based on random finite set (RFS) is an effective method to deal with multi-target tracking (MTT). However, the traditional GM-PHD filter cannot form a continuous track in the tracking process, and it is easy to produce a large number of redundant invalid likelihood functions in a dense clutter environment, which reduces the computational efficiency and affects the update result of target probability hypothesis density, resulting in excessive tracking error. Therefore, based on the GM-PHD filter framework, the target state space is extended to a higher dimension. By adding a label set, each Gaussian component is assigned a label, and the label is merged in the pruning and merging step to increase the merging threshold to reduce the Gaussian component generated by dense clutter update, which reduces the computation in the next prediction and update. After pruning and merging, the Gaussian components are further clustered and optimized by threshold separation clustering, thus as to improve the tracking performance of the filter and finally realizing the accurate formation of multi-target tracks in a dense clutter environment. Simulation results show that the proposed algorithm can form a continuous and reliable track in dense clutter environment and has good tracking performance and computational efficiency.

Keywords: multi-target tracking (MTT); random finite set (RFS); probability hypothesis density filter; gaussian mixture; label track maintenance

1. Introduction

Multi-target tracking (MTT) is a process of assigning the measured values to the targets, filtering them at the same time, and managing the tracks of multiple targets according to the time step [1,2]. RFS provides a unified and clear framework for MTT. Probability Hypothesis Density (PHD) [3] filtering has a good potential in solving target tracking under insufficient prior knowledge and an unknown number of targets. At present, it has been widely used in radar target tracking [4–6], computer vision [7], real-time positioning and map building [8], and group target tracking [9].

Traditional tracking algorithms, such as probabilistic data association (PDA) and multiple hypothesis tracking (MHT) [10–13], all transform the multi-target problem into a parallel single-target tracking problem by assigning measurements. The core of its processing method is data correlation, but when there are many targets and a large number of false alarm clutter, the correlation will bring a combination explosion and make the calculation amount increase exponentially. Correlation error and state estimation error are coupled and influence each other, resulting in large estimation error. PHD filtering technology avoids complex data association problems in the process of state estimation and can concentrate resources to deal with tracking problems. Thus, it has good application performance. Among them, Gaussian mixture (GM) and sequential Monte Carlo (SMC) are two important methods of PHD operation, which are called GM-PHD [14] and SMC-PHD [15], respectively. For the SMC single target algorithm, reference [16] proposed an adaptive model discrimination method, and does not need to be adjusted for specific problems, and has a better effect on the model uncertainty of discrete data. Reference [17]
does not use model selection but integrates model information in the SMC method and dynamically adjusts the sampling step through the posterior predictive ability of each model to adapt to the uncertain environment of the model. Reference [18] uses interactive parallel particle filters, each filter is for a different model, and the method is simple and flexible, which solves the joint problem of online tracking and detection of the current modality. For the GM multi-target algorithm, reference [19] proposed an improved probability hypothesis density filter to modify the weight of the target when the target was missed. Reference [20] proposed a modified GM-PHD filter, which corrected the prediction and update equation of the PHD filter, and reduced the problem of information loss caused by a missed detection. Reference [21] improved the filter performance in the case of continuously missed alarms by establishing a survival probability model and adaptively adjusting the confirmation probability. This article is researched on the basis of GM-PHD.

PHD filter has a rigorous mathematical theoretical basis, which can realize joint detection and tracking of targets. The structure is complete and clear, and the amount of calculation is small. PHD filter cannot identify the tracks of different targets in the tracking process. With the increase of targets or the approach of distances, the wrong judgment of tracks leads to the inability of the GM-PHD filter to track the targets that need attention. Therefore, for a PHD filter, the accurate distinction of tracks is the key to ensuring tracking performance [22–24].

In MTT, it is the premise of establishing the track on determining the identity of the target. The data association algorithm [25] proposed by Panta provided a unique label for each target and obtained the track of a single target through the state at each time and the association between the targets [14]. Vo et al. [26–28] also proposed label Multi-Bernoulli filter and label multi-target Bayesian processing method, which effectively solved the track formation problem of RFS processing multi-target, and belonged to the pioneering work of label RFS. Reference [29] used dynamic weight allocation based on detection to reduce the weight error of adjacent targets and proposed a novel tracking management scheme, which effectively solved the track recognition of adjacent targets [30]. In addition, Reference [30] proposed a track extraction method based on target topology information to identify targets and manage tracks.

In the background of dense clutter, the PHD filter is difficult to form a continuous and stable track and has a large tracking error. In the framework of GM-PHD filtering, this paper introduces the Gaussian component label set into the pruning and merging step and proposes a threshold separation clustering algorithm considering velocity and position information to extract the target state. This method not only outputs the target track accurately but also improves the tracking accuracy and reduces the computational complexity. The main contributions of this paper are as follows: (1) the merging and pruning step is improved, the Gaussian components with the same label are merged, and the continuous missing detection parameters are used for moderate track preservation. (2) A method of clustering Gaussian components by threshold separation is proposed. After pruning and merging, Gaussian components are clustered by velocity threshold and Euclidean distance. The filtering performance of the improved algorithm has been verified by simulation scenes.

The following chapters are arranged as follows: Section 2 describes the traditional PHD filtering, including PHD recursion, Gaussian mixture implementation, and label GM-PHD algorithm. Section 3 introduces the improvement of label GM-PHD filtering, including the merging and pruning steps of introducing label components, threshold separation clustering method, the formation of tracking associated tracks, and the comprehensive implementation of the algorithm. In Section 4, the effectiveness of the algorithm is verified by linear scene simulation experiments. Section 5 is the conclusion.
2. PHD Filtering

2.1. PHD Recursion

Considering the following MTT scenario, the target state set is $X_k = \{x_k^1, \ldots, x_k^{N_k}\}$ and the measurement set is $Z_k = \{z_k^1, \ldots, z_k^M\}$, where $x_k^n$ and $z_k^m$ represent the $n$-th target state and the $m$-th measurement at time $k$, respectively. $N_k$ and $M_k$ are the number of targets and the number of measurements at time $k$, respectively, and they change with time and the environment. Assuming that the prior probability of multi-target approximately obeys Poisson distribution, according to the RFS statistics theory, the PHD recurrence equation is [3]

$$D_{k|k-1}(x) = \int \left( p_{s,k|k-1|f_k|k-1}(x|\xi) \right) + \beta_{k|k-1}(x|\xi)D_{k-1|k-1}(\xi) d\xi$$

$$+ \gamma_k(x)$$

$$D_{k|k}(x) = [1 - P_{D,k}] D_{k|k-1}(x)$$

$$+ \sum_{z_k \in Z_k} \frac{P_{D,k}(z_k|x) D_{k|k-1}(x)}{\lambda c(z_k) + \int P_{D,k}(z_k|x) D_{k|k-1}(\xi) d\xi}$$

$\gamma_k(x)$ and $\beta_{k|k-1}(x|\xi)$ represent the target strength of newborn and spawned RFS respectively [31], $p_{s,k|k-1}$ is the survival probability of target at time $k$ on $k-1$, $P_{D,k}$ represents the detection probability of target at time $k$, $f_{k|k-1}(x|\xi)$ is the probability density function of state transition, $g_k(z|x)$ is the likelihood function of a single target, and $\lambda$ is the average number of clutter, and $c(z_k)$ is probability density functions of Poisson clutter distribution.

2.2. Gaussian Mixture Implementation

GM-PHD filter represents the prior PHD (1) and the posterior PHD (2) of multi-target as Gaussian mixture formation, and its iterative recursion can be expressed as a prediction update structure similar to Kalman filter (KF), and the algorithm is processed in a multi-target space.

Assume that the multi-target posterior PHD at time $k - 1$ can be expressed as a Gaussian mixture, and the equation is as follows:

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

where $l_{k-1}$ is the number of Gaussian components at time $k - 1$. In general, the number of $l_k$ is much greater than the number of $N_k$. $w_{k-1}^i$ is the weight of the $i$-th Gaussian component, $m_{k-1}^i$ and $P_{k-1}^i$ represents the mean and covariance of the $i$-th Gaussian component. Then time $k$ multi-target prediction PHD and time $k$ posterior multi-target PHD can be expressed as Gaussian mixture form:

$$D_{k|k-1}(x) = \gamma_k(x) + D_{\beta_{k|k-1}}(x)$$

$$+ D_{\gamma_k}(x)$$

$$= \sum_{i=1}^{l_k} w_{k|k-1}^i N(x; m_{k|k-1}^i, P_{k|k-1}^i)$$

$$= \sum_{i=1}^{l_k|k-1} w_{k|k-1}^i N(x; m_{k|k-1}^i, P_{k|k-1}^i)$$

$$m_{k|k-1}^i = Fm_{k-1}^i$$

$$P_{k|k-1}^i = Q_k + Fp_{k-1}^i F^T$$

$$D_k(x) = [1 - P_{D,k}] D_{k|k-1}(x)$$

$$+ \sum_{i=1}^{m_k} \sum_{j=1}^{l_k} w_k^i(z) N(x; m_{k}^i, p_{k}^i)$$
where $D_{\beta_{k-1}}(x)$ and $D_{\gamma_{k-1}}(x)$ are the Gaussian mixture intensity function of spawned target and surviving target respectively, $I_{k|k-1}$ is the number of Gaussian components of predicting PHD, and $w_{k|k-1}^j$, $m_{k|k-1}^j$ and $p_{k|k-1}^j$ are respectively the weight, mean, and covariance of the $i$-th Gaussian component in the prediction intensity function. $w_i^k$, $m_i^k$, and $p_i^k$ are, respectively, the weight, mean, and covariance of the $j$-th component in the update intensity function, $F$ is the state transition matrix, $H_k$ is the measurement transition matrix, $Q_k$ is the process noise covariance, $R_k$ is the measurement noise covariance matrix, and $I$ is the identity matrix.

The expected number of targets $\hat{N}_{k|k-1}$ and $\hat{N}_k$ associated with $D_{k|k-1}$ and $D_k$ are obtained as follows:

$$\hat{N}_{k|k-1} = \hat{N}_{k-1} \left( p_{\beta,k} + \sum_{j=1}^{I_{\beta,k}} w_{\beta,k}^j \right) + \sum_{j=1}^{I_{\gamma,k}} w_{\gamma,k}^j$$

$$\hat{N}_k = \hat{N}_{k|k-1} (1 - p_{D,k}) + \sum_{z \in Z_k} \sum_{j=1}^{I_{k|k-1}} w_k^j(z)$$

Among them, $I_{\beta,k}$ and $I_{\gamma,k}$ respectively, represent the number of spawned Gaussian components and newborn Gaussian components at time $k$.

Therefore, the terms required to represent $D_k$ at step $k$ is $I_k = \left( I_{k-1} (1 + I_{\beta,k}) + I_{\gamma,k} \right) (1 + |Z_k|)$. However, the number of Gaussian components and the computational complexity increases very fast.

### 2.3. Label GM-PHD Filter

In Reference [14], a tracking label was introduced into the GM-PHD filter, which extracts the target track by assigning a label to each Gaussian component. Its algorithm includes the following steps:

**Step 1:** initialize the Gaussian mixture posterior PHD at $t = 0$:

$$D_0(x) = \sum_{i=1}^{I_0} w_0^i N \left( x; m_0^i, p_0^i \right)$$

Assign a unique label to each Gaussian component:

$$\Gamma_0 = \{ \tau_0^1, \cdots, \tau_0^{I_0} \}$$

$\tau_0^i$ is label of the $i$-th Gaussian component, and the mean is $m_0^i$ and covariance is $p_0^i$.

**Step 2:** Predict according to Equation (4), the label set $\Gamma_{k|k-1}$ construction method is as follows:

The surviving Gaussian component still retains the original label, and the newborn or spawned Gaussian component assigns a newborn label, then the predicted label set is defined as:

$$\Gamma_k = \{ \tau_k^1, \cdots, \tau_k^{I_k} \}$$

$$w_k^i(z) = \frac{P_{D,k} w_{k-1}^i \delta_{k|k-1}^i \left( z; m_{k|k-1}^i, p_{k|k-1}^i \right)}{\lambda_c(z_k) + \sum_{j=1}^{I_{k|k-1}} w_{k|k-1}^j \delta_{k|k-1}^j \left( z; m_{k|k-1}^j, p_{k|k-1}^j \right)}$$

$$K_k^i = p_{k|k-1}^{i,j} H_k^T \left[ H_k p_{k|k-1}^{i,j} H_k^T + R_k \right]^{-1}$$

$$m_k^i = m_{k|k-1}^i + K_k^i \left( z_k^i - H_k m_{k|k-1}^i \right)$$

$$p_k^i = \left( I - K_k^i H_k \right) p_{k|k-1}^i$$
\[ \Gamma_{k|k-1} = \Gamma_{k-1} \cup \{ \tau_{b,k}^{1,1}, \ldots, \tau_{b,k}^{l_b,k} \} \cup \{ \tau_{i,k}^{1}, \ldots, \tau_{i,k}^{J_i,k} \} \]  
(16)

Among them, \( \tau_{b} \) is a spawned Gaussian component label and \( \tau_{i} \) is a newborn Gaussian component label.

Step 3: Updating is performed according to Equation (7), and all Gaussian terms obtained after updating are assigned the same labels as their previously associated prediction terms.

Step 4: Pruning and merging

\[ \{ \omega_{i,k}^{j}, m_{i,k}^{j}, p_{i,k}^{j} \}_{i=1}^{N} \] is the PHD Gaussian component set updated at time \( k \); Set a truncation threshold \( T_{r} \), a merging threshold \( T_{m} \), a maximum number of Gaussian terms \( J_{\text{max}} \), and two sets \( A \) and \( B \).

Gaussian components weighted greater than the truncation threshold are put into set \( A \):

\[ A = \{ i \in \{ 1, \ldots, J_{\text{max}} \} \mid \omega_{i,k}^{j} > T_{r} \} \]  
(17)

Then find out the index value of Gaussian term with the largest weight in set \( A \):

\[ j = \arg \max_{i \in A} \omega_{i,k}^{j} \]

and make set \( B \) as

\[ B = \left\{ i \in A \mid \left( m_{i,k}^{j} - m_{k}^{j} \right)^{T} \left( p_{i,k}^{j} \right)^{-1} \left( m_{i,k}^{j} - m_{k}^{j} \right) \leq T_{m} \right\} \]  
(18)

If the two Gaussian components satisfy Equation (18), they are combined into one term. The combined component parameters are obtained by weighted summation:

\[ \tilde{\omega}_{k}^{j} = \sum_{i \in B} \omega_{i,k}^{j} \]  
(19)

\[ \tilde{m}_{k}^{j} = \frac{1}{\tilde{\omega}_{k}^{j}} \sum_{i \in B} \omega_{i,k}^{j} m_{i,k}^{j} \]  
(20)

\[ \tilde{p}_{k}^{j} = \frac{1}{\tilde{\omega}_{k}^{j}} \sum_{i \in B} \omega_{i,k}^{j} \left( p_{i,k}^{j} + \left( \tilde{m}_{k}^{j} - m_{k}^{j} \right) \left( \tilde{m}_{k}^{j} - m_{k}^{j} \right)^{T} \right) \]  
(21)

Step 5: Multi-target state extraction

Besides extracting the target state, the set composed of Gaussian component labels is also extracted

\[ \hat{\Gamma}_{k} = \{ \tau_{i} \mid \omega_{i,k}^{j} > 0.5 \} \]  
(22)

\[ \hat{X}_{k} = \{ m_{i,k}^{j} \mid \omega_{i,k}^{j} > 0.5 \} \]  
(23)

Step 6: Track extraction

At each moment, if there is a Gaussian component or its spawned Gaussian term \( \omega_{i,k}^{j} > 0.5 \), it is marked as “confirmed” and other branches at \( \omega_{i,k}^{j} \leq 0.5 \) are marked as “tentative”.

3. Robust Label GM-PHD Filter

In a dense clutter environment, due to the existence of a large number of clutters, a large number of Gaussian components with unknown sources will appear in the pruning and merging step. If not properly processed, the computational complexity of the label GM-PHD filter will be greatly increased in the subsequent prediction and update steps. The existence of dense clutter will also affect the final state extraction of the filter and then directly affect the formation of correct multi-target tracks. It is necessary to consider improving the pruning and merging method, and on this basis, optimize the state extraction process of the algorithm, thus that the output track is correct and stable, and at the same time improve the efficiency of the algorithm.
3.1. Improve Pruning and Merging Methods

As mentioned earlier, GM-PHD filter uses the Gaussian mixture distribution to model multi-target to update the posterior strength of multi-target. However, such as other PHD filters, this filter does not consider the time sequence correspondence of Gaussian components in the tracking process. Thus, it is impossible to distinguish targets. Label GM-PHD filtering method solves this problem, which uses pruning and merging process to exclude unreal targets and ensure the formation of tracks. However, the pruning and merging method does not consider the label set of Gaussian components. When the target was occluded, or the distance was close, the Gaussian components corresponding to the real target were pruned or merged, which significantly degrades the tracking performance of the algorithm. The main reason why targets cannot be tracked when they are close to each other is that the label of the Gaussian component is ignored. If the merging threshold $T_m$ is set relatively large, the number of merged Gaussian components will increase, which will lead to an increase in the probability of mixing different labels, thus that the 2 targets cannot be effectively distinguished. If the merge threshold setting $T_m$ is small, it is not conducive to the aggregation of real targets. A large number of invalid Gaussian components are still retained, which greatly increases the subsequent calculation burden. The above problems can be solved by the following improvements, that is, only Gaussian terms with the same label were allowed to be merged. This processing was not only beneficial to the problem of multi-target discrimination in dense clutter environments but can also consider setting a larger merging threshold $T_m$ to improve the target tracking performance. The expression is as follows:

$$B = \left\{ i \in A \mid \left( m_i^k - m_j^k \right)^T \left( p_i^k \right)^{-1} \left( m_i^k - m_j^k \right) \leq T_m, \tau_i^k = \tau_j^k \right\}$$

The combined component parameters were obtained by Equations (19)–(21). In order to reduce the operation time of each step, reference [31] proposes a tree structure, in which the initial Gaussian component was set as the origin, and if there were Gaussian components or their spawned Gaussian terms $\omega^i_k > 0.5$, the tree was marked as “confirmed”, and other branches unsatisfied the $\omega^i_k > 0.5$ were marked as “tentative”. In the process of target tracking, due to the influence of clutter and noise, the target may disappear for several frames before appearing. The confirmed Gaussian component $n_{\text{missed}}$ was assigned a parameter representing the number of consecutive missed detections on the branch. The method allows previously confirmed components $n_{\text{missed}} \leq 3$ marked as tentative in the current step.

The proposed algorithm flow is as follows:

1. For $\{ \omega^i_k, m^i_k, p^i_k \}_{i=1}^{k}$, the truncation threshold $T_r$, the merging threshold $T_m$ and the maximum number of Gaussian terms $J_{\text{max}}$
2. Set the set $A = \{ i \in \{1, \cdots , J_{\text{max}}\} \mid \omega^i_k > T_r \}$
3. Find the index value of the maximum weight value in set $A$
   $$j = \arg \max_{i \in A} \omega^i_k$$
   Obtain the set $B = \left\{ i \in A \mid \left( m_i^k - m_j^k \right)^T \left( p_i^k \right)^{-1} \left( m_i^k - m_j^k \right) \leq T_m, \tau_i^k = \tau_j^k \right\}$
4. Weighted summation to obtain Gaussian component parameters
   $$\tilde{\omega}^i_k = \sum_{i \in B} \omega^i_k$$
   $$\tilde{m}^i_k = \frac{1}{\tilde{\omega}^i_k} \sum_{i \in B} \omega^i_k m^i_k$$
\[ \hat{p}_k^c = \frac{1}{\omega_k^c} \sum_{i \in B} \omega_k^i \left( p_i^k + (\hat{m}_k^c - m_k^i) (\hat{m}_k^c - m_k^i)^T \right) \]

5. Put component parameter \( \{\hat{\omega}_k^c, \hat{m}_k^c, \hat{p}_k^c\} \) into the output \( \text{Output}_k \)

\[ \text{Output}_k = \{ \text{Output}_k, \{\hat{\omega}_k^c, \hat{m}_k^c, \hat{p}_k^c\} \} \]

3.2. Threshold Separation Clustering

Based on the improve pruning and merging methods, GM-PHD extracts Gaussian components with weight greater than 0.5 after pruning and merging as output and retains the remaining Gaussian components to continue iterative filtering. Due to the persistent interference of strong clutter in a dense clutter environment, it is easy to produce wrong Gaussian terms with large weights. If they are extracted and regarded as newborn targets, it will lead to high tracking errors.

It can be seen that finding the real target in a dense clutter environment and effectively separating the real target from the wrong Gaussian term are the key to solving the problem. Reference [32] proposed a threshold separation clustering method. On this basis, this paper proposes a method of clustering Gaussian components of the same label using velocity and distance information and extracting the target state, which can effectively aggregate the real target state and prevent redundant error. The Gaussian term interferes with the extraction of the target state and can reduce the amount of calculation on the basis of [32].

Firstly, the index value of the Gaussian component with the largest weight in set \( B \) was found, and the weight value was placed in the set \( S_{\omega} \).

\[ j = \arg \max_{n=1:N} (\omega_n^p) \]  
\[ S_{\omega} = S_{\omega} \cup \{ \omega_j^p \} \]

where \( N \) represents the number in set \( B \) after pruning and merging.

The velocity difference and the corresponding Euclidean distance between the Gaussian component with the largest weight and the Gaussian component with the same label in set \( B \) were calculated.

\[ \text{speed}_{i,j} = \left( \left( m_k^i - m_k^j \right) \ast R_0 \right) \left( p_k^i \right)^{-1} \left( \left( m_k^i - m_k^j \right) \ast R_0 \right)^T, \quad \tau_k = \tau_k^j \]  
\[ d_{i,j} = \sqrt{\left( m_k^i \ast R_1 - m_k^j \ast R_2 \right)^2 + \left( m_k^i \ast R_1 - m_k^j \ast R_2 \right)^2}, \quad \tau_k = \tau_k^j \]

wherein \( R_0 \) is used for the velocity information for separating, \( R_1 \) and \( R_2 \) is used for the position information for separating the target. In a linear system, \( R_0 \) is column matrices \([0 1 0 0]^T\), and \( R_1 \) and \( R_2 \) are column matrices \([1 0 0 0]^T\) and \([0 0 1 0]^T\), respectively.

Finally, all Gaussian components satisfying the given velocity threshold \( M_{\text{speed}} \) and Euclidean distance \( M_d \) are clustered.

\[ S_{g,k} = \left\{ x_k^i \in B \mid \text{speed}_{i,j} \leq M_{\text{speed}} \& \& d_{i,j} \leq M_d \right\} \]  
\[ S_k = S_k \cup \{ S_{g,k} \} \]

wherein the set \( S_k \) contains all Gaussian components satisfying the conditions of Equations (27) and (28), and the clustering result is used as the final target state estimation.
3.3. Tracking Correlation to Form a Track

In the Gaussian component set $S_k$ obtained by threshold separation clustering, the Gaussian component with the largest weight in the corresponding label was found, which can effectively estimate the target track and introduce a track set $\mathcal{Z}_k$.

$$\mathcal{Z}_k = \{ r \mid r^i \in \tau^i_k \}$$ (31)

$$\mathcal{Z}_k = \{ r^i_k \} \cup \mathcal{Z}_{k-1}$$ (32)

wherein $r^i_k$ is the i-th track corresponding to time $k$, and $r^i_k$ represents the corresponding track labeled $\tau^i_k$. If there is a track with a label $\tau^i_k$ in the set $\mathcal{Z}_{k-1}$ and its parameter $n_{missed} = 0$, $m^i_k$ would be added to the labeled $\tau^i_k$ track and added to the track set. This Gaussian component is confirmed to match the tracking label $\tau^i_k$.

If there is a track in the set $\mathcal{Z}_{k-1}$ and has a label $\tau^i_k$, but it is not added to the track set $\mathcal{Z}_k$, the number of missed detections is judged:

$$\begin{cases} 
\mathcal{Z}_k^i = \{ r \mid r^i_k \in \tau^i_k, n_{missed} \leq 3 \} \\
M^i_k = F_k m^i_{k|k-1}
\end{cases}$$ (33)

If Gaussian component $n_{missed} \leq 3$, $m^i_k$ would be added to the labeled track $\tau^i_k$.

Finally, multi-target state estimation can be obtained:

$$\hat{X}_k = \{ m^i_k : \tau^i_k \in \hat{\Gamma}_k, i = 1, \ldots, N \}$$ (34)

3.4. Overall Steps to Improve the Algorithm

Aiming at the problem of poor tracking performance and low computational efficiency of traditional label GM-PHD in dense clutter environment, a robust label GM-PHD algorithm was proposed. The overall steps were as follows:

Step 1: Initialize the parameters

When $t = 0$, a Gaussian mixture posterior PHD and an initial label are set, such as Equations (35) and (36):

$$D_0(x) = \sum_{i=1}^{J_0} w^i_0 N(x; m^i_0, p^i_0)$$ (35)

$$\Gamma_0 = \{ \tau^1_0, \ldots, \tau^J_0 \}$$ (36)

Each Gaussian component is recursive by the Bayesian criterion.

Step 2: Predict step

The Gaussian component is predicted according to Equation (4), and then newborn labels are assigned to the newborn and spawned Gaussian components, and the newborn label set and the spawned label set are added to the predicted label set:

$$\Gamma_{k|k-1} = \Gamma_{k-1} \cup \{ \tau^{1,1}_{\beta k}, \ldots, \tau^{J_{\beta k}}_{\beta k}, \tau^{1,1}_{\gamma k}, \ldots, \tau^{J_{\gamma k}}_{\gamma k} \}$$ (37)

Step 3: Update the steps

According to Equation (7), the prediction state is updated using the measurements at time $k$, and then all updated Gaussian terms are assigned the same label as their previously associated prediction terms.

Step 4: Pruning and merging

A state-updated Gaussian mixture PHD $\{ \omega^i_k, m^i_k, p^i_k \}_{i=1}^{J_k}$ is obtained; set a truncation threshold $T_r$, a merging threshold $T_m$, a maximum number $J_{max}$ of Gaussian terms, and two sets $A$ and $B$. 
The Gaussian component satisfying the truncation threshold was found and put into the set $A$, and then the Gaussian term with the largest weight $j = \arg\max_{i \in A} \omega_k^{(i)}$ in the set $A$ was found, and the Gaussian term was used for screening and merging:

$$B = \left\{ i \in A \mid \left( m_k^i - m_k^j \right)^T \left( p_k^i \right)^{-1} \left( m_k^i - m_k^j \right) \leq T_m, \tau_k^i = \tau_k^j \right\}$$

The combined component parameters were obtained by weighted summation:

$$\tilde{\omega}_k^i = \frac{1}{\omega_k^j} \sum_{i \in B} \omega_k^i$$

$$\tilde{m}_k^i = \frac{1}{\omega_k^j} \sum_{i \in B} \omega_k^i m_k^i$$

$$\tilde{p}_k^i = \frac{1}{\omega_k^j} \sum_{i \in B} \omega_k^i \left( p_k^i + \left( \tilde{m}_k^i - m_k^j \right) \left( \tilde{m}_k^i - m_k^j \right)^T \right)$$

The Gaussian component of Set $A$ that occurs in both Set $A$ and Set $B$ was deleted until Set $A$ was an empty set.

**Step 5: Threshold separation clustering**

A velocity threshold and an Euclidean distance threshold were set, and Gaussian components satisfying Equations (43) and (44) and having the same label were clustered, and then Gaussian components satisfying conditions were put into a set $S_k$, and the clustering criteria were as follows:

$$\text{speed}_{i,j} = \left( \left( m_k^i - m_k^j \right) \times R_0 \right) \left( p_k^i \right)^{-1} \left( \left( m_k^i - m_k^j \right) \times R_0 \right)^T$$

$$d_{i,j} = \sqrt{\left( m_k^i \times R_1 - m_k^j \times R_2 \right)^2 + \left( m_k^i \times R_1 - m_k^j \times R_2 \right)^2}$$

**Step 6: Tracking association to form track**

If there is a track with a label $\tau_k^i$ in the set $\mathcal{S}_{k-1}$ and its parameters $n_{missed} = 0$, the corresponding Gaussian term $m_k^i$ is added to the labeled track $\tau_k^i$, and it is added to the track set $\mathcal{S}_k$, at which time the Gaussian component is confirmed to match the tracking label $\tau_k^i$.

If there is a track in the set $\mathcal{S}_{k-1}$ with a label $\tau_k^i$, but it is not added to the track set $\mathcal{S}_k$, the number of missed detections should be judged:

$$\left\{ \begin{array}{l} \xi_k^i = \{ r | r_k^i \in \tau_k^i, n_{missed} \leq 3 \} \\ m_k^i = E_k m_k^i_{k|k-1} \end{array} \right.$$  (45)

If Gaussian component $n_{missed} \leq 3$, $m_k^i$ would be added to the labeled track $\tau_k^i$.

**4. Simulation and Experimental Results**

**4.1. Experimental Parameter Setting**

In order to verify the effectiveness of the proposed algorithm in MTT scenarios, the improved label GM-PHD filter was compared with GM-PHD filter, label GM-PHD filter, and Gaussian mixture cardinalized probability hypothesis density (GM-CPHD) [28] filter. The experimental parameters were set as follows.

The tracking scene was set to multiple targets in four possible locations or spawned from other targets, and the observation area was $[-1000, 1000] \times [-1000, 1000]$ (m$^2$). There
were 12 batches of targets in the scene. For the sake of simplicity, it was assumed that each target moves in a straight line at a uniform speed.

The state vector of the target consists of position and velocity components: \( x_k = [p_{x,k} \ p_{y,k} \ v_{x,k} \ v_{y,k}] \), and its state equation is:

\[
\begin{bmatrix}
1 & 0 & T & 0 \\
0 & 1 & 0 & T \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_{k-1} \\
T^2/2 \\
0 \\
T^2/2 \\
0 \\
T
\end{bmatrix}
\begin{bmatrix}
x_k \\
T \\
0 \\
T
\end{bmatrix}
\begin{bmatrix}
w_k-1 \\
\end{bmatrix}
\]

\( w_k \sim N(0, 5) \).

The sampling interval \( T \) is 1 s, the total tracking time was 100 s, and the noisy process is \( w_k \sim N(0, 5) \). The intensity of newborn target was as follows (47):

\[
\gamma_k(x) = \sum_{i=1}^{4} \omega_i \cdot \gamma_i \cdot m_i \cdot \gamma_i \cdot p_i \cdot \gamma_i
\]

Among them, \( m_{1,k} = [400, -10, -600, 5]^T \), \( m_{2,k} = [-800, 20, -200, -5]^T \), \( m_{3,k} = [-200, 15, 800, -5]^T \), \( m_{4,k} = [0, -20, 0, -15]^T \), and weight of newborn target \( \omega_i = 0.03 \). The process noise of newborn targets obeys Gaussian distribution, the mean value was zero, and the covariance was:

\[
Q = \text{diag}(100, 100, 100, 100).
\]

The measurement vector is position information:

\[
z_k = \begin{bmatrix} p_{x,k} \ p_{y,k} \end{bmatrix}
\]

and the measurement equation is given by the following equation:

\[
z_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix} x_k + v_k
\]

where measured noise was \( v_k \sim N(0, 5) \). Clutter is evenly distributed in the moving area of the target, with average scanning \( \lambda \). In the Gaussian component pruning section, the truncation threshold of the Gaussian component is \( 10^{-5} \). The state extraction threshold was set to 0.5. The merging threshold was set to 10, the maximum number of Gaussian components was 100, the velocity threshold \( M_{\text{speed}} = 10 \), and the Euclidean distance \( M_d = 40 \). The number of Monte Carlo simulations was 100. Evaluating tracking quality by OSPA distance:

\[
\text{OSPA}_{p,c}(x_k, \hat{x}_k) = \sqrt{p \left( \sum_{i=1}^{x_k} \left( d_e \left( x_k^i, \hat{x}_k \right) \right)^p + c^p \left( |\hat{x}_k| - |x_k| \right) \right)}
\]

\( x_k \) is the target state vector, the two parameters of OSPA distance were set to \( p = 1 \) and \( c = 200 \), respectively. The smaller the OSPA distance, the higher the accuracy of target state estimation.

4.2. Simulation Scenario 1

In this MTT scene, clutter rate \( \lambda = 90 \), survival probability \( p_{s,k} = 0.99 \), and detection probability \( p_{D,k} = 0.95 \). Figure 1 is the real track and actual tracking measurement map of multiple targets in dense clutter and high detection probability environment. As can be seen from Figure 1b, in the MTT scene with dense clutter, the proposed improved GM-PHD algorithm can accurately obtain the target track. In Figure 1b, we can see that there are false tracks, which are caused by the clutter around the measured values of newborn targets. They will affect the stability of the filter before the track is formed. However, after clustering by speed and distance, the influence would gradually disappear with time, and the wrong target track would not appear.
\[ k^x \] is the target state vector, the two parameters of OSPA distance were set to \( D_{kp} = 200 \), \( \lambda = 90 \). Figure 1 is the real track and actual tracking measurement: (a) True target trajectory; (b) Improved GM-PHD filter tracking results.

4.2.1. Filter Quality Analysis

Figure 2 compares the performance of various filters in dense clutter and high detection probability environment, including GM-PHD filter, label GM-PHD filter, robust label GM-PHD filter, and GM-CPHD filter. Obviously, the tracking performance of the labeled GM-PHD filter proposed in this paper was obviously better than GM-PHD, traditional labeled GM-PHD, and GM-CPHD in dense clutter environments. As can be seen from Figure 2a,b, the OSPA distance between GM-PHD and label GM-PHD filters is large, which shows that the dense clutter environment seriously affects the filter performance. As can be seen from Figure 2c,d, the improved labels GM-PHD and GM-CPHD have better performance in estimating the number of targets because the improved label GM-PHD filter combines Gaussian components with the same label and adds a threshold clustering separation step after merging and pruning, while GM-CPHD filter calculates the second moment of a posterior probability density function at the expense of operation time. Thus it is more accurate to estimate the number of multiple targets.

4.2.2. Computational Complexity Analysis

In the process of pruning and merging the traditional GM-PHD filter and the label GM-PHD filter, all Gaussian components were pruned and merged. The algorithm proposed in this paper only processes the Gaussian components with the same label. It will undoubtedly greatly reduce the amount of calculation of the Gaussian component and avoid the problem of exponential increase in the amount of calculation. Moreover, the algorithm proposed in this paper also includes a threshold separation clustering method, which also processes Gaussian components with the same label. It can be seen in the following simulation that processing the Gaussian components with the same label will reduce the computational complexity very well.

Figure 3 compares the running time of different filters in this scenario. It can be seen that the improved label GM-PHD filtering algorithm improves computational efficiency. Compared with the GM-PHD filtering algorithm, it has slightly higher computational complexity but compared with the other two filtering algorithms, it has higher computational efficiency because it uses label sets to combine Gaussian components with the same label, thus reducing the computational complexity. Figure 3 also shows that the improvement of GM-CPHD tracking performance is at the expense of computing time to some extent.
Compared with the GM-PHD filter, the operation time of the improved label GM-PHD filter is still 13.14% higher than that of the GM-PHD filter, but the computational efficiency is improved by 11.23% compared with the traditional label GM-PHD filter.

Figure 2. Performance comparison of different algorithms in dense clutter and high detection probability environment: (a) OSPA distance comparison; (b) OSPA Location comparison; (c) target cardinalities estimation error; (d) estimated cardinalities of targets.

Figure 3. Different algorithm running time.
4.3. Simulation Scenario 2

In this MTT scene, clutter rate $\lambda = 90$, survival probability $p_{sk} = 0.99$, and detection probability $p_{D,k} = 0.8$. Figure 4 shows the tracking performance comparison of each filter in dense clutter and low detection probability environment. As can be seen from Figure 4, except for the improved label GM-PHD filtering algorithm, the tracking performance of other filtering algorithms is obviously degraded under the condition of low detection probability, which indirectly proves the robustness of the proposed algorithm. After pruning and merging, the Gaussian components are separated and clustered by using velocity and distance thresholds, which effectively improves the tracking performance of multi-targets under the environment of low detection probability.

4.4. Simulation Scenario 3

In this MTT scene, the survival probability $p_{sk} = 0.99$, detection probability $p_{D,k} = 0.9$, the proposed algorithm under different clutter rate tracking performance superiority was verified by simulation. Figure 5 shows the performance comparison of algorithms with different clutter rates. As can be seen from Figure 5a–c, the tracking errors of the GM-PHD filter and the label GM-PHD filter increases linearly with the increasing clutter rate. The reason is that with the increasing clutter, the Gaussian component pruning and combining step becomes difficult to handle, resulting in error coupling of various errors. The improved
label GM-PHD filtering algorithm uses the Gaussian component of the same label to process the subsequent steps, which effectively avoids these problems. It can be seen from Figure 5d that with the increase of clutter rate, the calculation amount of GM-CPHD increases linearly, while the calculation amount of the improved label GM-PHD filtering algorithm changes little, which also reflects the stability of the algorithm.

![Figure 5](image_url)

**Figure 5.** Comparison of the performance of different filters in a changing environment of clutter rate: (a) OSPA distance comparison; (b) OSPA Location comparison; (c) target cardinalities estimation error; (d) running time comparison.

4.5. Simulation Scenario 4

In this MTT scene, the survival probability $p_{s,k} = 0.99$ and clutter rate $\lambda = 60$ were simulated to prove the performance advantages of the proposed algorithm in different detection probability environments. Figure 6 shows the comparison of algorithm performance under different detection probabilities. It can be seen from Figure 6 that the improved label GM-PHD filtering algorithm had higher stability than other algorithms and had certain advantages in tracking performance under different detection probabilities.
Simulations of the above four scenarios show that the improved label GM-PHD filtering algorithm has good tracking performance in dense clutter and low detection probability environment. It also has good stability in the environment where clutter rate and detection probability change.

This algorithm is based on GM-PHD and focuses on multi-target tracking in a dense clutter environment, and the motion of the target is also linear. In the next work, we can study non-linear high-maneuvering multi-targets and Multi-sensor observation scenes, using SMC algorithm for non-linear moving targets. By considering the tracking problems in the case of model uncertainty, the algorithms [16–18] can be more suitable for actual tracking scenarios.

5. Conclusions

Aiming at the problem of low accuracy and low computational efficiency of MTT in dense clutter environments, this paper proposes an efficient and accurate robust label GM-PHD filtering algorithm. Based on the traditional label GM-PHD, this algorithm improves the pruning and merging step to merge Gaussian components with the same label, which greatly reduces the computational burden and error merging in a dense clutter environment. According to velocity threshold and Euclidean distance, the newborn state
extraction method can effectively reduce the occurrence of false tracks and improve the performance of MTT in dense clutter environments. The proposed algorithm can be further improved to adapt to MTT in an unknown parameter environment.

**Author Contributions:** K.W. performed the simulations and wrote the paper; Q.Z. and X.H. revised this paper and offered some useful suggestions in methodologies and simulations. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**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.

**References**

1. Reid, D. An Algorithm for Tracking Multiple Targets. *IEEE Trans. Autom. Control.* 1979, 24, 1202–1211. [CrossRef]

2. Daum, F. Multitarget-Multisensor Tracking: Principles and Techniques. *IEEE Aerosp. Electron. Syst. Mag.* 1996, 11, 41–53. [CrossRef]

3. Mahler, R. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* 2004, 39, 1152–1178. [CrossRef]

4. Hu, Q.; Ji, H.; Zhang, Y. A standard PHD filter for joint tracking and classification of maneuvering extended targets using random matrix. *Signal Process.* 2017, 144, 352–363. [CrossRef]

5. Leonard, M.R.; Zoubir, A.M. Multi-target tracking in distributed sensor networks using particle PHD filters. *Signal Process.* 2019, 159, 130–146. [CrossRef]

6. Liu, Z.; Ji, L.; Yang, F.; Qu, X.; Yang, Z.; Qin, D. Cubature Information Gaussian Mixture Probability Hypothesis Density Approach for Multi Extended Target Tracking. *IEEE Access.* 2019, 7, 103678–103692. [CrossRef]

7. Feng, P.; Wang, W.; Dlay, S.; Naqvi, S.M.; Chambers, J. Social Force Model-Based MCMC-OCSVM Particle PHD Filter for Multiple Human Tracking. *IEEE Trans. Multimed.* 2016, 19, 725–739. [CrossRef]

8. Kim, H.; Granström, K.; Gao, L.; Battistelli, G.; Kim, S.; Wymeersch, H. Joint CKF-PHD Filter and Map Fusion for 5G Multi-cell SLAM. In Proceedings of the IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6.

9. Jiang, Q.; Wang, R.; Zhou, C. Modified Bayesian Group Target Track Initiation Algorithm Based on Algebraic Graph Theory. *J. Electron. Inf. Technol.* 2021, 43, 531–538.

10. Bar-Shalom, Y. Tracking and Data Association; Academic Press: San Diego, CA, USA, 1988.

11. Bar-Shalom, Y.; Kirubarajan, T.; Lin, X. Probabilistic data association techniques for target tracking with applications to sonar, radar and EO sensors. *IEEE Aerosp. Electron. Syst. Mag.* 2005, 20, 37–56. [CrossRef]

12. Musicki, D.; Evans, R. Joint integrated probabilistic data association: JIPDA. *IEEE Trans. Aerosp. Electron. Syst.* 2004, 40, 1093–1099. [CrossRef]

13. Blackman, S.S. Multiple hypothesis tracking for multipletarget tracking. *IEEE Aerosp. Electron. Syst. Mag.* 2004, 19, 5–18. [CrossRef]

14. Panta, K.; Clark, D.E.; Vo, B.N. Data Association and Track Management for the Gaussian Mixture Probability Hypothesis Density Filter. *IEEE Trans. Aerosp. Electron. Syst.* 2009, 45, 1003–1016. [CrossRef]

15. 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.

16. Drosvandi, C.C.; McGree, J.; Pettitt, A.N. A sequential Monte Carlo algorithm to incorporate model uncertainty in Bayesian sequential design. *J. Comput. Graph. Stat.* 2014, 23, 3–24. [CrossRef]

17. Urteaga, I.; Bugallo, M.F.; Djuric, P.M. Sequential Monte Carlo methods under model uncertainty. In Proceedings of the IEEE Statistical Signal Processing Workshop (SSP), Palma de Mallorca, Spain, 26–29 June 2016; p. 15.

18. Martinos, L.; Read, J.; Elvira, V.; Louzada, F. Cooperative Parallel Particle Filters for on-Line Model Selection and Applications to Urban Mobility. *Digit. Signal Process.* 2017, 60, 172–185. [CrossRef]

19. Xi, S.; Feng, Y.; Yan, L.; Quan, P. Improved probabilistic hypothesis density filter with low detection probability. *Command. Inf. Syst. Technol.* 2014, 5, 36–40.

20. Liu, Z.X.; Xie, W.X.; Wang, P.; Yu, Y. A Gaussian mixture PHD filter with the capability of information hold. *Acta Electron. Sin.* 2013, 41, 1603–1608.

21. Yazdian-Dehkhordi, M.; Azimifar, Z. Refined GM-PHD tracker for tracking targets possible subsequent missed detections. *Signal Process.* 2015, 116, 112–126. [CrossRef]

22. Clark, D.E.; Bell, J. Data association for the PHD filter. In Proceedings of the 2005 International Conference on Intelligent Sensors, Melbourne, VIC, Australia, 5–8 December 2005; pp. 217–222.
23. Zhou, X.; Tang, Y.; Yang, J.; Xie, Z.; Chen, S. Penalized Gaussian mixture probability hypothesis density tracker with multi-feature fusion. In Proceedings of the 2014 IEEE International Conference on Robotics and Biomimetics, Bali, Indonesia, 5–10 December 2014; pp. 1415–1420.

24. Wang, Y.; Jing, Z.; Hu, S. Data association for PHD filter based on MHT. In Proceedings of the 2008 11th International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; pp. 1–8.

25. Panta, K.; Vo, B.N.; Singh, S. Novel data association schemes for the probability hypothesis density filter. *IEEE Trans. Aerosp. Electron. Syst.* 2007, 43, 556–570. [CrossRef]

26. Reuter, S.; Vo, B.T.; Vo, B.N.; Dietmayer, K. The labeled multi-Bernoulli filter. *IEEE Trans. Signal Process.* 2014, 62, 3246–3260.

27. Vo, B.T.; Vo, B.N. Labeled random finite sets and multi-object conjugate priors. *IEEE Trans. Signal Process.* 2013, 61, 3460–3475. [CrossRef]

28. Vo, B.N.; Vo, B.T.; Phung, D. Labeled random finite sets and the Bayes multi-target tracking filter. *IEEE Trans. Signal Process.* 2014, 62, 6554–6567. [CrossRef]

29. Zhang, H.; Yang, J.; Ge, H.; Yang, L. An improved GM-PHD tracker with track management for multiple target tracking. In Proceedings of the 2015 International Conference on Control, Automation and Information Sciences (ICCAIS), Changshu, China, 29–31 October 2015; pp. 185–190.

30. Yang, F.; Su, Y.; Zhang, W.; Yao, X. A track extraction method based on target topology for PHD filter. In Proceedings of the 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), Wuhan, China, 11–13 November 2016; pp. 449–453.

31. Yang, F.; Wang, Y.; Liang, Y.; Pan, Q. Overview of multi-target tracking technology based on probability hypothesis density filtering method. *Acta Autom. Sin.* 2013, 39, 1944–1956. [CrossRef]

32. Luo, Q.; Gao, Z.; Xie, C. Improved GM-PHD filter based on threshold separation clusterer for space-based starry-sky background weak point target tracking. *Digit. Signal Process.* 2020, 103, 102766. [CrossRef]