Research Article

Yang Lei, Yuan Wu*, and Ahmad Jalal Khan Chowdhury

Multi-target tracking algorithm in intelligent transportation based on wireless sensor network

https://doi.org/10.1515/phys-2018-0121
Received Oct 04, 2018; accepted Nov 14, 2018

Abstract: The traditional extended Kalman algorithm for multi-target tracking in the field of intelligent transportation does not consider the occlusion problem of the multi-target tracking process, and has the disadvantage of low multi-target tracking accuracy. A multi-target tracking algorithm using wireless sensors in an intelligent transportation system is proposed. Based on the dynamic clustering structure, the measurement results of each sensor are the superimposed results of sound signals and environmental noise from multiple targets. During the tracking process, each target corresponds to a particle filter. When the target spacing is relatively close to each other, each master node realizes distributed multi-target tracking through information exchange. At the same time, it is also necessary to consider the overlap between adjacent frames. Since the moving target speed is too fast, the target occlusion has the least influence on the tracking accuracy, and can accurately track multiple targets. The experimental results show that the proposed algorithm has a target tracking error of 0.5 m to 1 m, and the tracking result has high precision.

Keywords: Wireless sensor network, intelligent traffic field, multi-target tracking, algorithm research, node, distributed tracking

PACS: 07.07.Mp, 07.05.Kf, 89.20.Ff

1 Introduction

Over the years, with the development and maturity of technologies such as wireless communications, integrated circuits, sensors, and micro-electro mechanics, mass production of low-cost, low-power, multi-function miniature sensors has become possible. A wireless sensor network (WAN) is a multi-hop self-organizing network system formed by radio communication in a large number of micro sensor nodes deployed in the monitoring area [1]. One of the important applications of WSN is target location and tracking. At present, in the field of single-target tracking of WSN, some excellent research results have appeared, as in the reference, such as extended Kalman (EKF) algorithm and parallel particle filtering algorithm (PEPF). But there is relatively little research on WSN multi-target tracking. Among the latter, the existing algorithms mainly use the traditional multi-target tracking method, which uses the multi-target tracking algorithm based on the association hypothesis to solve the WSN multi-target tracking problem [2]. In addition, some scholars have conducted in-depth research on multi-target tracking without considering data association based on the weak detection capability of WSN node signals.

In the multi-target tracking of traffic in a WSN-based intelligent transportation system, the energy, sensing range, communication ability and computing power of nodes in WSN are limited [3], the authors of [4] adopted the traditional centralized algorithm. In [5], the computing power of the target node is poor, and there is a poor tracking accuracy of the multi-target result. So, the algorithm achieved the tracking of the target from the motion process of the intelligent traffic domain target. However, due to the high number of targets in the road traffic and the high similarity between them, there were many target tracking errors. In order to solve the shortcomings of these algorithms, a large number of distributed algorithms have been proposed in recent years. Distributed algorithms have a high application value in WSN target tracking because they can reduce energy consumption and extend network lifetime [5]. Therefore, this paper proposes a multi-target intelligent transportation tracking algorithm based on a
wireless sensor network to achieve effective tracking of multiple targets in road traffic.

2 Research on multi-target intelligent transportation tracking algorithms based on wireless sensor networks

2.1 WSN Network model

The wireless sensor network consists of S acoustic sensor nodes, and all nodes are evenly distributed in the monitoring area. The position of each sensor node is known and it can detect the surrounding sound intensity [6]. At the same time, wireless communication is possible between the sensor nodes, and each node knows the location of its neighbor nodes (distance is less than the communication distance).

2.2 Multi-target motion model in intelligent transportation systems

Assume K targets move in the monitoring area [7], where K is known to be fixed. The motion between the targets is relatively independent, and the state of the k-th target at time t is

\[ X_{k,t} = \begin{bmatrix} x_{k,t}, y_{k,t}, \dot{x}_{k,t}, \dot{y}_{k,t} \end{bmatrix}, \quad k = 1, 2, \ldots, K \]

\[ X_{k,t} = f(X_{k,t-1}, w_{k,t-1}) \]

where \( f(\cdot) \) is a state transfer function, which can be a linear or nonlinear function [8], \( w_{k,t-1} \) is the state noise of the k-th target and its probability distribution is known.

2.3 Measurement model

Assume all sensors are acoustic sensors, that the observation of the sensor is the superposition of sound signals from multiple targets with ambient noise [9], and the observations of the individual sensors are independent of each other. Then, at time t, the energy of the sound signal received by the sensor \( s \) at position \( r_s = [x_s, y_s] \) is:

\[ y_{s,t} = g_s(X_t) + v_{s,t} = \sum_{k=1}^{K} \frac{\Psi_k}{\|r_s - I_{k,t}\|^2} + v_{s,t} \]

(2)

where \( g_s(\cdot) \) represents the sound energy function from the target measured by the s-th node [10]: \( X_t = [X_{1,t}, X_{2,t}, \ldots, X_{K,t}]^T \) is the joint state of K targets; \( \Psi_k \) is the sound energy from the k-th target measured at the unit distance; \( I_{k,t} = [x_{k,t}, y_{k,t}] \) is the position of the target k at time t:

\[ \|r_s - I_{k,t}\|^2 = (x_s - x_{k,t})^2 + (y_s - y_{k,t})^2 \]

(3)

Equation (3) is the Euclidean distance between \( r_s \) and \( I_{k,t} \).

In equation (2), \( \alpha \) is the loss factor, which is determined by the propagation medium of the signal and can be considered to be constant [11]. \( \alpha \) is equal for all sensors. \( v_{s,t} \) is the sensor receives the noise intensity at the time t, and it is independent of \( w_{k,t}, k = 1, 2, \ldots, K \). When the time window length T for averaging energy is sufficiently long, \( v_{s,t} \) can be considered to obey a normal distribution with a mean of \( u_v \) and a variance of \( \sigma_v^2 \).

2.4 Distributed multi-target tracking algorithm

In WSN, the node density is generally large. If all nodes are always active, then node resources are wasted [12], and this easily leads to communication conflicts. Therefore, the node is generally in a dormant state, and only when it receives an activation message from another node, does it become active and begins to participate in the observation. In addition, it can be seen from the observation equation of the sensor [13] that when the target distance is far away, the signal received by the sensor from the target is small, resulting in a small signal-to-noise ratio, and the observation information has little effect on the update of the target state [14]. Therefore, in the process of tracking, at each moment we select the appropriate nodes to participate in the observation around the target according to the state information of each target. When there are multiple targets in the WSN monitoring area and if the targets are far apart, the signals of the other targets received by the sensors near a single target are weak and can be ignored [15], so only single target tracking is performed. Only when the targets are close together so the signals received by the sensors near the target are stronger [16], then the measurement is considered to be a signal from multiple targets, which requires multi-target joint tracking. Therefore, we need to establish certain rules to determine whether to conduct multi-target joint tracking [17]. Based on the above analysis, the basic idea of our distributed multi-target tracking algorithm is: one master node is selected for each target at a time, and the particle filter corresponding to the target is run by the master node. The master node selects appropriate slave nodes to par-
participate in the observation according to certain rules, and judges the distance between the targets according to certain rules. If a target is far away from other targets, only single target tracking is performed [18]. If a target is closer to some other targets, the target master node exchanges target state prediction information with each other, and each master node uses multiple state information of the target to calculate the particle weight of the current target to realize distributed multi-target tracking [19]. When multiple targets are jointly tracked, the master nodes corresponding to the respective targets exchange state information by wireless communication.

2.4.1 Selection of master and slave nodes

In the initial stage, we can specify the node closest to each target as its master node according to the a priori information of each target [20]. In the process of tracking, the current master node selects the node closest to the predicted position of the target as the master node of the next moment according to the predicted state of the target.

At each moment, the master node of each target chooses a certain slave node to participate in the observation [21]. Different rules can be used when selecting slave nodes.

2.4.2 Multi-target joint tracking rules

Joint tracking is required between multiple targets only when the distances between the targets are relatively close and affect each other. Since it is distributed tracking, the state information of each target is stored on different master nodes, so it is impossible to directly judge the distance between the targets. According to the selection of the master and slave nodes above, the slave nodes are all in the vicinity of the master node [22]. Therefore, we can use the master node to have a common slave node to determine the distance between the targets. The specific idea is as follows: the master node selects the slave node according to the defined rules. When a slave node receives the message of more than two master nodes at the same time, i.e., receives the message of the master nodes \( S_1, S_2, \ldots, S_N \), the slave node sends the information of node \( S_j, j = 1, 2, \ldots, N \) and \( j \neq i \) to the node \( S_i \), so that node \( S_i \) knows that the target on the master node is to be jointly tracked with the targets on the other N-1 master nodes. If the master node does not receive information related to the target of the other master node, the target corresponding to the master node performs single target tracking.

2.4.3 Distributed multi-target tracking process

Distributed multi-target tracking is to find the correspondence between each target in the preceding and succeeding frames in consecutive frames. In smart city traffic monitoring, the monitored moving target speed is generally less than 100 km/h, it is monitored by a camera with a frame rate of 30 frames/s, the displacement of the target is less than 1 m between adjacent frames. This has the following characteristics in the sequence of adjacent images collected [23]: (1) The same target has a smaller average grayscale difference between two adjacent frames; (2) There must be overlapping areas between adjacent frames of the same target, so that the absolute value of the area difference of the same target is the minimum.

Using these characteristics, three eigenvalues are determined: centroid distance \( L \), average gray level \( H \), and area difference \( A \), respectively, as shown in the following equations:

\[
L_{ij} = \frac{\sqrt{(x_i^t - x_j^{t+1})^2 + (y_i^t - y_j^{t+1})^2}}{\max_n \sqrt{(x_i^t - x_n^t)^2 + (y_i^t - y_n^t)^2}}
\]

(4)

\[
H_{ij} = \frac{|G_i^t - G_{t+1}^t|}{\max_n |G_i^t - G_{t+1}^t|}
\]

(5)

\[
A_{ij} = \frac{|S_i^t - S_{t+1}^t|}{\max_n |S_i^t - S_{t+1}^t|}
\]

(6)

The above three equations represent the relationship between the target values in the current frame and the respective feature values of the target in the next frame, where \((x, y)\) represents the centroid coordinate of the target, \(G\) is the gray scale of the target, \(A\) is the area of the target, and the defined feature vector is \(V = (L, H, A)^T\). The distance measure function between targets is:

\[
D^{mn} = V^{mn}W
\]

(7)

where, \(W\) is a diagonal 3*3 matrix, representing the weight \((w_1, w_2, w_3)\) of each element of the feature vector, \(m\) and \(n\) are the sequence numbers of the targets in adjacent frames, and the weight of the features is not fixed [24]. If we see the multi-objective matching as a global optimization problem, then the aim is to determine the overall optimum match for each target in the adjacent frame by finding the
minimum value of $D_T$:

$$D_T = \sum_{m=1}^{k} D_{mn}$$  \hspace{1cm} (8)

where $k$ is the number of targets in the next frame. When a Lagrange multiplier is introduced, $D_T$ is:

$$D_T = \sum_{m=1}^{k} D_{mn} + \lambda(w_t w_h - 1)$$  \hspace{1cm} (9)

Therefore, the minimum value of $D_T$ can be solved by:

$$\begin{cases}
\frac{\partial D_T}{\partial w_t} = 0; \frac{\partial D_T}{\partial w_h} = 0; \frac{\partial D_T}{\partial \lambda} = 0 \\
w_t w_h - 1 = 0
\end{cases}$$  \hspace{1cm} (10)

Solving Eq. (10) gives:

$$\begin{align*}
w_t &= \left( \sum_{m=1}^{k} H_{mn} \sum_{m=1}^{k} A_{mn} \right) \sqrt{\frac{\sum_{m=1}^{k} L_{mn}}{\sum_{m=1}^{k} H_{mn}^2}} \\
w_h &= \left( \sum_{m=1}^{k} L_{mn} \sum_{m=1}^{k} A_{mn} \right) \sqrt{\frac{\sum_{m=1}^{k} H_{mn}}{\sum_{m=1}^{k} A_{mn}^2}} \\
w_t &= \left( \sum_{m=1}^{k} H_{mn} \sum_{m=1}^{k} L_{mn} \right) \sqrt{\frac{\sum_{m=1}^{k} A_{mn}}{\sum_{m=1}^{k} L_{mn}^2}}
\end{align*}$$  \hspace{1cm} (11)

In order to find the best match, firstly, calculate the corresponding weight value according to Eq. (11). Then calculate $D_{mn}$, the minimum $D_{mn}$ is taken and its corresponding area $R^m$ is the best match of $R^n$. Defining a threshold of 5, if the minimum value $D_{mn}$ is greater than $a$, means that the target was not matched in the next frame, and the centroid is at the edge of the image. This situation corresponds to the target disappearing from the subsequent image. If the target is matched [25–31], there are still redundant targets in the matched target, and the centroid is at the edge of the image. This situation corresponds to a new target appearing in the image. If the new target just enters the image, then it will be in the matching process of the next cycle. The new target is inserted into the target list, and if a target leaves the screen, the target is removed from the target list.

### 2.4.4 Target occlusion

In the traffic scene, there is the problem of mutual occlusion between the moving vehicles. Due to the position of the camera, the projection of various vehicles on the road onto the picture sometimes forms a phenomenon of mutual occlusion or adhesion. Under ideal conditions, if the camera lens is facing straight down towards the center of the road and the height is sufficient, there will be no occlusion in the picture. However, in actual engineering installations, these ideal conditions are often not met. When installing a camera, it is often subject to various factors, so there is bound to be occlusion in practical applications. However, in the traffic scene, most of the occlusion is partial and short-term. Also, the occurrence of occlusion often means there are more vehicles on the road, and so the vehicles generally do not perform large maneuvers, which means the occluded vehicle is trackable. The occlusion problem reflects the phenomenon of target sticking in the image. In the continuous frame image, the process of “merging” or “separating” is as follows:

$$\begin{cases}
R_{ij} \rightarrow R_{(t+1)k} \\
R_{ih} \rightarrow R_{(t+1)k}
\end{cases}$$  \hspace{1cm} (12)

wherein, Eq. (12) expresses a “Merging” case, in which multiple targets in the current frame find the same target in the next frame to match the best one and Eq. (13) expresses a “Separating” case, in which case a mismatch occurs. Since the target in the current frame matches the target in the next frame, it is only possible to get a minimum value of $D_{mn}$, so that the intelligent system gets a best matching target, and for the other targets that are separated, a mismatch occurs. For the separation phenomenon, the search method can be determined by some constraints of the urban traffic monitoring system: In the next frame, according to the characteristics that the separation targets are not too far apart, other targets are searched within a certain range of the centroid distance of the mismatched target (the centroid distance threshold $T$ is set), and the matching targets in the current frame are found. According to the characteristics of the separation target, the target areas in the front and back frames must have intersections, and the mismatched targets are intersected with the matching targets of other targets. The target with the largest intersection area is found, that is, the matching target of the mismatched target is found.

### 3 Results

In order to verify the effectiveness of the proposed algorithm, we performed Monte Carlo simulations in Matlab. Supposing that $S$=100 acoustic sensors are evenly distributed in an area of 200 m x 200 m. The origin of the coordinates is in the lower left corner of the coverage area of
the wireless sensor network (Figure 2). The sampling period $T_{\text{period}} = 1$ s of all sensors is measured from the unit distance. The signal strength of the target is $\psi = 500$, the loss factor is $\alpha = 2$, the observed noise is $\delta = 0.01$, and the communication distance of the node is $C=30$ m. There are three targets for uniform motion in the monitoring area. The initial position of target 1 is (40 m, 160 m) and the speed is (2 m/s, -2 m/s). The initial position of target 2 is (40 m, 120 m), and the speed is (2 m/s, 0). The initial position of target 3 is (40 m, 20 m) and the speed is (2 m/s, 1 m/s). The system state noise of all three targets is $Q_w = \text{diag} \{[0.01, 0.01]\}$.

For the simulation, 100 Monte Carlo simulations were run, the number of particles is $N=1000$, the initial state of the target obeys the normal distribution, and the initial state information of target 1 is $x_{1,0} = [40, 160, 2, -2]^T$, $\Xi_{1,0} = \text{diag} \{[1, 1, 0.01, 0.01]\}$. The initial state information of target 2 is $x_{2,0} = [40, 120, 2, 0]^T$, $\Xi_{2,0} = \text{diag} \{[1, 1, 0.01, 0.01]\}$. The initial state information of target 3 is $x_{3,0} = [40, 20, 2, 1]^T$, $\Xi_{3,0} = \text{diag} \{[1, 1, 0.01, 0.01]\}$. In addition, the simulation is mainly to verify the feasibility of the distributed structure of the algorithm. Therefore, in the simulation process, when selecting the slave nodes, the neighbor node rules are used to reduce the number of calculations, and the target tracking accuracy is estimated by the target position with the root mean square error (RMSE). The simulation results are shown in Figures 1 to 4:

- **Figure 1**: Comparison of the real and estimated target trajectories

  It can be seen from the simulation results in Figures 1 to 3 that the distributed multi-target tracking algorithm can effectively track three targets with high tracking accuracy.

  Figure 4 shows the number of nodes that are active at each moment during the simulation. It can be seen from the simulation results that due to the dynamic clustering structure, only the nodes around the target are activated at each moment. The relatively centralized algorithm requires all nodes to be activated, and the number of active nodes is greatly reduced, thereby reducing energy consumption and thus prolonging network life.

  Figure 5 compares the amount of data that is required to be communicated by the extended Kalman (EKF) algorithm, the parallel particle filter algorithm (PEPF), and the
Multi-target tracking algorithm in intelligent transportation field

The number of activation nodes proposed algorithm. It can reflect the network consumption under different algorithms.

As can be seen from Figure 5, tracking of multiple targets in an intelligent transportation system requires a certain amount of data to be communicated. However, it can be seen that the average data traffic of the proposed algorithm is lower than that of the EKF algorithm and the PEPF algorithm, which is good.

Based on the above experiments, comparing the average communication data volume results of the three algorithms, multiple experiments are performed to compare the error of the target tracking distance and the target tracking time. The experiment randomly selects 12 road segments in the intelligent transportation system in a certain area, and multiple targets for target tracking in each road segment are randomly selected. Figure 6 and 7 show the tracking distance error and target tracking time results of the three algorithms under multiple experiments.

Figure 6: Target tracking distance error results of three

Analysis of the data in Figure 6 shows that there is a certain degree of tracking distance error in multi-target tracking in the intelligent transportation system. It can be seen from the data in the analysis table that the target tracking error of the EKF algorithm in the experimental test fluctuates between 111.520 m and 224.3 m; the target tracking error value of the PEPF algorithm varies from 124.61 m to 194.62 m. The target tracking error distance of the proposed algorithm fluctuates between 52.35 m and 71.25 m. Comparing the results, the target tracking accuracy of the proposed algorithm is high, and the effect of target tracking is good.

Algorithms

The efficiency of tracking the target, i.e., the time taken, is the key to representing the effectiveness of the algorithm. The analysis of the results in Figure 7 shows that the time difference between different algorithms for multi-target tracking is large. The results show that the target tracking times for the EKF algorithm and the PEPF algorithm is consistently much higher than the times of the proposed algorithm, which indicates that the proposed algorithm has strong applicability in improving target tracking efficiency.
4 Discussion

From the simulation results in Figures 1 to 4, for time $1s \leq T \leq 28s$, target 1 and 2 are jointly tracked. For $29s \leq t \leq 39s$, target 1 should be jointly tracked with target 3, and target 2 should be jointly tracked with target 1 and target 3 is also required to be jointly tracked with target 1. For $40s \leq t \leq 45s$, target 1 is to be tracked jointly with target 2 and 3. Target 2 is tracked with target 1 and 3, and target 3 is tracked jointly with targets 1 and 2. For $46s \leq t \leq 50s$, target 1 should be tracked jointly with target 3. Target 2 should be tracked jointly with target 3. Target 3 should be tracked jointly with target 1 and 2. It can be seen that in the execution of the algorithm in this paper, the algorithm can automatically determine whether the target needs joint tracking according to the distance between the targets, thus realizing true distributed multi-target tracking. The proposed algorithm realizes the tracking of multi-target in an intelligent transportation system mainly because a wireless sensor network can contain a large number of installed micro sensor nodes in the intelligent traffic monitoring area and form a multi-hop self-organizing network system through radio communication, thus providing support for the accurate tracking of the target algorithm.

Analysis of the results of Table 1 and Figure 6 shows that the target tracking accuracy of the proposed algorithm is far superior to the other two algorithms, and the error is small. The algorithm presented in this paper fully considers the problems of coincidence, occlusion and a high speed of the target being tracked. The matching distance measure function is established by using the distance between the target in two adjacent frames, the average gray level of the target and the area difference. The minimum distance determines the best matched. The results of Table 2 and Figure 7 show that the proposed algorithm takes significantly less time for multi-target tracking than the other two algorithms. The main reason is that only the nodes in the active state can perform target tracking, which reduces the number of nodes to be tracked, thus greatly reducing the computation time.

### Table 1: The tracking error results of three algorithms under many experiments

<table>
<thead>
<tr>
<th>Experimental number</th>
<th>EKF algorithm (m)</th>
<th>PEPF algorithm (m)</th>
<th>This paper algorithm (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>111.52</td>
<td>165.23</td>
<td>62.35</td>
</tr>
<tr>
<td>2</td>
<td>130.235</td>
<td>187.35</td>
<td>54.76</td>
</tr>
<tr>
<td>3</td>
<td>203.112</td>
<td>165.432</td>
<td>71.25</td>
</tr>
<tr>
<td>4</td>
<td>195.262</td>
<td>152.354</td>
<td>68.35</td>
</tr>
<tr>
<td>5</td>
<td>213.25</td>
<td>166.45</td>
<td>54.35</td>
</tr>
<tr>
<td>6</td>
<td>125.6</td>
<td>135.6</td>
<td>66.59</td>
</tr>
<tr>
<td>7</td>
<td>136.21</td>
<td>124.61</td>
<td>54.62</td>
</tr>
<tr>
<td>8</td>
<td>146.2</td>
<td>177.65</td>
<td>52.35</td>
</tr>
<tr>
<td>9</td>
<td>152.643</td>
<td>159.8</td>
<td>57.65</td>
</tr>
<tr>
<td>10</td>
<td>142.68</td>
<td>160.47</td>
<td>54.85</td>
</tr>
<tr>
<td>11</td>
<td>224.3</td>
<td>184.66</td>
<td>59.68</td>
</tr>
<tr>
<td>12</td>
<td>210.33</td>
<td>194.62</td>
<td>64.37</td>
</tr>
<tr>
<td>Average distance</td>
<td>165.95</td>
<td>164.51</td>
<td>60.09</td>
</tr>
</tbody>
</table>

### Table 2: Results of target tracking using three algorithms in many experiments

<table>
<thead>
<tr>
<th>Experimental number</th>
<th>EKF algorithm (s)</th>
<th>PEPF algorithm (s)</th>
<th>This paper algorithm (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>23.35</td>
<td>24.62</td>
<td>14.52</td>
</tr>
<tr>
<td>2</td>
<td>22.16</td>
<td>27.35</td>
<td>13.26</td>
</tr>
<tr>
<td>3</td>
<td>26.38</td>
<td>24.65</td>
<td>11.23</td>
</tr>
<tr>
<td>4</td>
<td>24.35</td>
<td>23.54</td>
<td>15.21</td>
</tr>
<tr>
<td>5</td>
<td>21.62</td>
<td>24.26</td>
<td>13.21</td>
</tr>
<tr>
<td>6</td>
<td>30.25</td>
<td>25.86</td>
<td>14.17</td>
</tr>
<tr>
<td>7</td>
<td>28.6</td>
<td>27.56</td>
<td>12.36</td>
</tr>
<tr>
<td>8</td>
<td>26.67</td>
<td>32.21</td>
<td>12.35</td>
</tr>
<tr>
<td>9</td>
<td>32.31</td>
<td>24.67</td>
<td>14.35</td>
</tr>
<tr>
<td>10</td>
<td>26.54</td>
<td>32.55</td>
<td>11.03</td>
</tr>
<tr>
<td>11</td>
<td>33.62</td>
<td>24.87</td>
<td>10.28</td>
</tr>
<tr>
<td>12</td>
<td>25.46</td>
<td>36.2</td>
<td>10.69</td>
</tr>
<tr>
<td>Mean time</td>
<td>26.77</td>
<td>27.36</td>
<td>12.72</td>
</tr>
</tbody>
</table>
5 Conclusions

Due to the limited resources of wireless sensor networks, traditional centralized multi-target tracking algorithms are limited in multi-target tracking using wireless sensor networks. This paper, proposes a distributed multi-target tracking algorithm based on the dynamic clustering structure of the acoustic sensor network. This algorithm uses the principle that the intensity of the sound signal is inversely proportional to the propagation distance. Only the appropriate nodes are selected around the target to participate in the tracking, and multi-target joint tracking is determined according to the target distance. In the process of tracking, each target corresponds to a particle filter. When the targets are far apart, single target tracking is performed. When the targets are relatively close to each other, each master node realizes distributed multi-target tracking through information exchange.

In the special scenario of urban traffic monitoring, monitoring equipment characteristics and vehicle motion characteristics form some of the constraints of the monitoring environment. For these constraints, the adaptive background method is used to extract the different moving targets and mark them. The matching distance measure function is established by using the distance between the targets in two adjacent frames, the average gray level of the target, and the area difference. By finding the global optimal solution, the distance measure is minimized, and then the best matching target is determined. The occlusion phenomenon in the target motion process was analyzed and a solution is proposed. The results show that the proposed algorithm can effectively achieve distributed tracking of multiple targets and ensures a high tracking accuracy.

Acknowledgement: Key projects of the National Social Science Foundation of China - (No.16AGJ007); Chongqing Science and Technology Commission special service platform for science and technology - Capacity building of mountain digital agriculture R & D Service Center (No.cstc2015ptfw-ggfw80001).

References


