IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 26, 2018, accepted January 11, 2019, date of publication January 24, 2019, date of current version February 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2894641

Measurement Data Fusion Based on Optimized Weighted Least-Squares Algorithm for Multi-Target Tracking

XINXIN HE[1,2], TAO WANG[3], WEI LIU[4], AND TAO LUO[1,2], (Senior Member, IEEE)

[1]Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China
[3]School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
[4]State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Tao Luo (tluo@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871041 and Grant 61571065, and in part by the China Postdoctoral Science Foundation under Grant 2017M620695.

ABSTRACT The outliers remove, the classification of effective measurements, and the weighted optimization method of the corresponding measurement are the main factors that affect the positioning accuracy based on range-based multi-target tracking in wireless sensor networks. In this paper, we develop an improved weighted least-square algorithm based on an enhanced non-naive Bayesian classifier (ENNBC) method. According to the ENNBC method, the outliers in the measurement data are removed effectively, dataset density peaks are found quickly, and remaining effective measurements are accurately classified. The ENNBC method improves the traditional direct classification method and took the dependence among continuous density attributes into account. Four common indexes of classifiers are used to evaluate the performance of the nine methods, i.e., the normal naive Bayesian, flexible naive Bayesian (FNB), the homologous model of FNB (FNB_{ROT}), support vector machine, k-means, fuzzy c-means (FCM), possibilistic c-means, possibilistic FCM, and our proposed ENNBC. The evaluation results show that ENNBC has the best performance based on the four indexes. Meanwhile, the multi-target tracking experimental results show that the proposed algorithm can reduce the root-mean-squared error of the position compared with the extended Kalman filter. In addition, the proposed algorithm has better robustness against large localization and tracking errors.

INDEX TERMS Range-based multi-target tracking, wireless sensor networks, weighted least-square, naive Bayesian, localization root mean squared error.

I. INTRODUCTION

Wireless sensor network (WSN) have been applied in many domains, such as smart home, intelligent transportation, and intelligent computing technology and so on [1]. The multi-target localization and tracking [2] are the important research contents of WSN [3]–[8]. Although the advantages of WSN (such as low cost, easy deployment and long-term work) bring new prospective for positioning applications, the characteristics of sensor nodes with susceptible to interference, poor reliability, random distribution, and communication distance [9]–[12] also present great challenges to localization in WSN. Usually, most of the tracking algorithms based on WSN can be grouped into two categories: range-free localization (easily implemented and suitable for large-scale deployment) and range-based localization (higher accuracy) [13]. This paper focuses on the research of range-based multi-target tracking technology. For the closely spaced multi-target tracking, the accuracy of the localization [14] is significantly affected by the processing algorithm of measurement data. Therefore, it is of great significance to research the processing algorithm of measurement data to improve the accuracy of the localization [15]–[17].

In recent years, the research interest of range-based multi-target localization and tracking has turned to the tracking problem of closely spaced targets. The processing algorithms of the classical measurement data for target tracking, such as Kalman filter (KF) [18], track association algorithm based on fuzzy comprehensive function [19], classical least-squares (CLS) algorithm [20], and Bayesian algorithm [21] etc, have no ability to identify the attributes of every measured data.

The optimal estimate values are calculated by the predictive values and the observation values based on most of the classical tracking algorithms in the process of range-based closely spaced multi-target localization and tracking [22], which are not accurate and is interfered by the measurement data of closely spaced targets. In the actual case, a large amount of measurement data is received from the sensor network, but it is not clear whether the measurement data belongs to a specific target, and measurement data of all targets is mixed with noise. The main work of this paper is to identify and classify measurement data by improving existing classification algorithms. The improved classification algorithm can distinguish the measurement data to the corresponding target for data fusion, and the noise can be filtered out. Therefore, the accuracy of the range-based closely spaced target tracking can be improved by the classification algorithm [23].

Some new clustering algorithms have been further developed in recent years, such as object tracking and credal classification with kinematic data in a multi-target context [24], non-naive Bayesian classifiers for classification problems with continuous attributes [25], resident location-recognition algorithm using a Bayesian classifier in the PIR sensor-based indoor location-aware system [26], a novel adaptive possibilistic clustering algorithm [27], a novel measurement data classification algorithm based on SVM for tracking closely spaced targets [28], real-time superpixel segmentation by DBSCAN clustering algorithm [29], measurement data classification optimization based on a novel evolutionary kernel clustering algorithm for multi-target tracking [13], a big data clustering algorithm for mitigating the risk of customer churn [30], a cloud-friendly RFID trajectory clustering algorithm in uncertain environments [31], a collaborative fuzzy clustering algorithm in distributed network environments [32], the differences between Bayesian classifiers and mutual-information classifiers [33], knowledge fusion for probabilistic generative classifiers with data mining applications [34], clustering by fast search and find of density peaks [35], k-means clustering with outlier removal [36], maximum margin Bayesian network classifiers [37]. In addition, the classification results of measurement data can be improved based on the weight analysis of classification process. References [38]–[40] show that better classification results can be obtained by using weighted data.

In this paper, a measurement data fusion method is designed to improve the distance-based close-range multi-target tracking performance in WSNs. A new ENNBC algorithm is proposed, which improves the traditional direct classification method and introduces the dependence between continuous density attributes into wireless sensor networks. First of all, a large number of outliers are removed from the measurement dataset and the density peak of every target measurement data is found quickly. Secondly, the ENNBC algorithm is applied to accurately classify remaining effective measurements. Thirdly, the weight values of each effective measurement are optimized by probability factors, and the optimal data centers of each target are calculated by weighted least-squares. Finally, the estimated location of every target at current time is obtained according to weighted least-square algorithm. Table 1 indexes the main mathematical symbols used in the description of the document.

TABLE 1. Main mathematical symbols.

x	n-dimensional state vector of the target
z	m-dimensional measurement vector of the sensor node
Z_k	Observation dataset at time k
f	Prediction model
g	Observation model
$\tilde{z}_k$	Measurement dataset at time k
$\hat{x}_k$	State estimate of the target at time k
$\mathbb{E}$	Expectation operator
θ	Sensor rotation angle
r	Measured distance between the sensor node and the target
ν	Class
c	Number of the class
P	Corresponding probability
$x^{(i)}$	Measurement data in the i-th class
$\mu_j^{(i)}$	Mean value in $x^{(i)}$
$\sigma_j^{(i)}$	Variance of all elements in $x^{(i)}$
K	Kernel function
$h_j^{(i)}$	Bandwidth parameter
ε	Neighborhood parameter
ρ_k	Density of measurement data
$\eta_k(i)$	Judgment condition of the density peak
$\tilde{x}$	Effective measurement data
$\hat{w}$	Weight vector
μ_i	Expected value of $\nu(\tilde{x})$ in i-th class
τ	Bias parameter
$w_j^{(i)}$	weighted value for $\tilde{x}_j^{(i)}$
$\hat{w}_j^{(i)}$	Normalized weighted value
Λ	Diagonal weight matrix

The structure of the paper is as follows. In Section II, the formulation of tracking problem, the method for position determination, CLS, and NBC are discussed in a Cartesian coordinate. In Section III, the novel outliers remove method, the ENNBC and weight value optimization of WLF are explained. In Section IV, we describe the hardware experimental platform of WSN and two Scenes. Section V describes the simulation results of the experiment in the WSN hardware platform. Section VI summarizes this paper.

II. PRELIMINARIES

The measurement data of the multi-target is collected through nodes in WSN, and is analyzed by several related methods. In this section, the problem formulation, the method for position determination, and CLS are discussed in a Cartesian coordinate. The section is divided into three parts: problem formulation, position determination method, and CLS analysis.

A. PROBLEM FORMULATION

The state variable is an effective method to describe the dynamic system [41]. With this method, the relationship

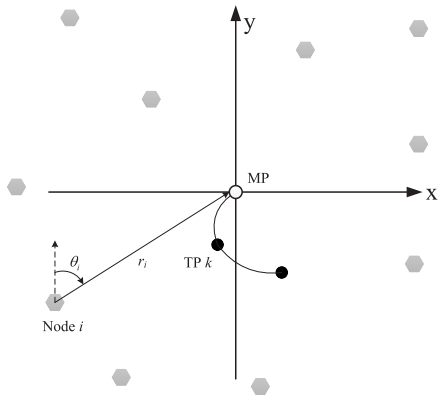


FIGURE 1. The geometric relationship between the target and sensor nodes.

between the input data and output data of the system can be discussed in time domain by the state transition model and output observation model. The output is a function of the state, which is usually disturbed by the random observation error [42]. When multiple targets enter the observation area, the sensor nodes begin to collect a large amount of measurement data, and a dataset is formed at each moment. The range-based multi-target tracking system is considered to be defined by geometry relationship between multiple moving targets and nodes. The black points (TP) in Fig. 1 are true position of the moving target j; Due to each target is homogenous, the subscript j is removed. k and $k-1$ are the corresponding time of the moving target. When the moving target moves to the MP position, n sensor nodes observe it, and receive the measured data at this time. At this time, position (x_{si}, y_{si}) of the i-th sensor node, the direction angle θ_i, and the measured distance ρ_i are obtained. The measured position of the moving target from i-th sensor node target is as follows:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_{si} + \rho_i \cdot \sin\theta_i \\ y_{si} + \rho_i \cdot \cos\theta_i \end{bmatrix} \tag{1}$$

A new observation dataset Z_k for the target will be obtained at time k. However, the uncertainty of maneuvering target movement from time $k-1$ to k should be considered. For the sensor node i, the expression of multi-target localization and tracking is as follows:

$$\begin{aligned} x_k &= f(x_{k-1}) + v_{k-1} \\ z_{i,k} &= g_i(x_k) + w_{i,k} \end{aligned} \tag{2}$$

where $x_k \in \mathbb{R}^n$ is the n-dimensional state vectors of the multi-target localization and tracking system at time k, $f(\cdot)$ is an unknown and possibly function of the states. $z_{i,k} \in \mathbb{R}^m$ is the m-dimensional measurement vectors of the sensor node i at time k. $\mathbb{R}^n$ and $\mathbb{R}^m$ are the n and m-dimensional real space, respectively. $g_i(\cdot)$ is a known and possibly function of the m-dimensional measurement vectors of the sensor i, v_k and $w_{i,k}$ are independent of each other with added zero meaning Gaussian measurement noise [43]. Because each sensor is homogenous, the subscript i is removed. $\tilde{z}_k = [z_{1,k}, z_{2,k}, \cdots, z_{n,k}]$ denote the measurement dataset of the target at time k [13]. The mission of target tracking in WSNs is to reduce the minimum-mean-squared-error (MMSE) estimator based on measurement dataset of each target, the expression is as follows:

$$\begin{aligned} MMSE(x_k) &= \sum_{x_{i,k} \in \mathbb{R}^n} P\left(x_{i,k} \,|z_k\right) \hat{x}_k \\ &= \sum_{x_{i,k} \in \mathbb{R}^n} P\left(x_{i,k} \,|z_k\right) \mathbb{E}\left(x_k \,|z_k, x_{i,k}\right) \end{aligned} \tag{3}$$

where $\mathbb{E}$ is the expectation operator; $\mathbb{E}\left(x_k|z_k, x_{i,k}\right)$ is the state estimate of the tracking system according to the measurement dataset of the target.

B. POSITION DETERMINATION ANALYSIS

For multi-target tracking, the mathematical model is established at first according to our range-based sensor network. The model includes the predicted and measured positions of the target, and the sensor position. Then the deviation between the predicted position and the measured value is calculated at time k, the position of the sensor i and the measurement position of the target, as shown in Fig. 2.

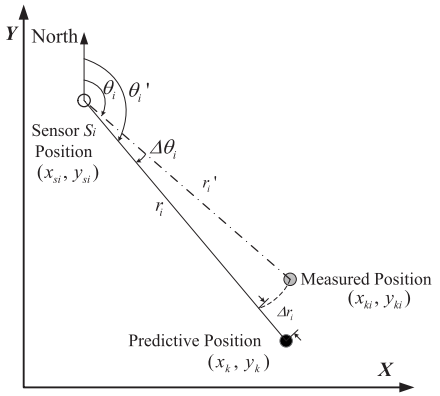


FIGURE 2. The geometric relationship between the prediction position, measured position, and nodes i.

The predicted position of the target is $(x_k,\ y_k)$ at time k, and the coordinate of the sensor S_i is $(x_{si},\ y_{si})$. The measurement position of the sensor S_i is $(x_{ki},\ y_{ki})$ from the target, and the following equations are obtained:

$$\begin{cases} \Delta x_{ki} = x_k - x_{ki} = r_i' \sin(\theta_i') - r_i \sin(\theta_i) \\ \Delta y_{ki} = y_k - y_{ki} = r_i' \cos(\theta_i') - r_i \cos(\theta_i) \end{cases} \tag{4}$$

where $(r_i,\ \theta_i)$ represents the measured value of the sensor S_i in the polar coordinate; $(r_i',\ \theta_i')$ represents the distance and angle between the sensor and the predicted position of the target in the polar coordinate. The equations for measurement and measurement error are as follows:

$$\begin{cases} r_i = r_i' - \Delta r_i \\ \theta_i = \theta_i' - \Delta\theta_i \end{cases} \tag{5}$$

where $(\Delta r_i,\ \Delta\theta_i)$ represents the error of the predictive value of the sensor S_i. According to the geometric relationship of Fig. 2, the equation is obtained as follows:

$$\begin{cases} \Delta x_{ki} = (x_k - x_{si}) - r_i \sin\theta_i \\ \Delta y_{ki} = (y_k - y_{si}) - r_i \cos\theta_i \end{cases} \tag{6}$$

using (6), the corresponding Δx_{ki} and Δy_{ki} of the n_s measurement data of the sensors were calculated at time k. The matrix of the Δx_{ki} and Δy_{ki} is defined as follows:

$$\begin{aligned} \boldsymbol{\Omega}_k &= (\boldsymbol{v}_{kx}, \boldsymbol{v}_{ky})^T \\ &= (\varepsilon_{k1}, \varepsilon_{k2}, \cdots \varepsilon_{kn_s}) \\ &= \begin{bmatrix} \Delta x_{k1} & \Delta x_{k2} & \cdots & \Delta x_{kn_s} \\ \Delta y_{k1} & \Delta y_{k2} & \cdots & \Delta y_{kn_s} \end{bmatrix} \end{aligned} \tag{7}$$

using (7) the distance $\Delta \boldsymbol{r}_{ki} = \|\varepsilon_{ki}\|_2$ is calculated. Then, using (4) and (5), the following equation is calculated:

$$\begin{cases} \Delta x_{ki} = (r_i + \Delta r_i)\sin(\theta_i + \Delta\theta_i) - r_i \sin\theta_i \\ \Delta y_{ki} = (r_i + \Delta r_i)\cos(\theta_i + \Delta\theta_i) - r_i \cos\theta_i \end{cases} \tag{8}$$

first, Δx_{ki} is calculated according to the above equation:

$$\begin{aligned} \Delta x_{ki} = {} & r_i \sin\theta_i \cos\Delta\theta_i + r_i \cos\theta_i \sin\Delta\theta_i \\ & + \Delta r_i \sin\theta_i \cos\Delta\theta_i + \Delta r_i \cos\theta_i \sin\Delta\theta_i - r_i \sin\theta_i \end{aligned} \tag{9}$$

in the actual target position detection, the error is less than the measurement, $\Delta r_i \ll r_i$ and $\Delta\theta_i \ll \theta_i$. The value of $\Delta\theta_i$ tends to be zero, and $\cos\Delta\theta_i \approx 1$, $\sin\Delta\theta_i \approx \Delta\theta_i$. Therefore, equation (9) can be simplified as:

$$\Delta x_{ki} \approx r_i \Delta\theta_i \cos\theta_i + \Delta r_i \sin\theta_i + \Delta r_i \Delta\theta_i \cos\theta_i \tag{10}$$

because the high-order item has little effect on the results, equation (10) is further simplified as:

$$\Delta x_{ki} \approx r_i \Delta\theta_i \cos\theta_i + \Delta r_i \sin\theta_i \tag{11}$$

similarly, Δy_{ki} is calculated as:

$$\Delta y_{ki} \approx -r_i \Delta\theta_i \sin\theta_i + \Delta r_i \cos\theta_i \tag{12}$$

using (11) and (12), the following equation is calculated:

$$\begin{cases} \Delta x_{ki} \sin\theta_i + \Delta y_{ki} \cos\theta_i = \Delta r_i \\ \Delta x_{ki} \dfrac{\cos\theta_i}{r_i} - \Delta y_{ki} \dfrac{\sin\theta_i}{r_i} = \Delta\theta_i \end{cases} \tag{13}$$

Note that (13) depends on the direction angle θ_i and the distance r_i of the node relative to the measurement.

C. CLS ANALYSIS

In order to perform it more conveniently, we use matrix algebra for analysis, so that (13) can be simplified as:

$$H\boldsymbol{\delta} = \boldsymbol{\xi} \tag{14}$$

where,

$$H = \begin{bmatrix} \dfrac{\sin\theta_1}{r_1} & \dfrac{\cos\theta_1}{r_1} \\ \dfrac{\cos\theta_1}{r_1} & -\dfrac{\sin\theta_1}{r_1} \\ \vdots & \vdots \\ \dfrac{\sin\theta_{m_k}}{r_{m_k}} & \dfrac{\cos\theta_{m_k}}{r_{m_k}} \\ \dfrac{\cos\theta_{m_k}}{r_{m_k}} & -\dfrac{\sin\theta_{m_k}}{r_{m_k}} \end{bmatrix}, \quad \boldsymbol{\delta} = \begin{bmatrix} \Delta x_k \\ \Delta y_k \end{bmatrix},$$

$$\boldsymbol{\xi} = \begin{bmatrix} \Delta\rho_1 \\ \Delta\theta_1 \\ \vdots \\ \Delta\rho_{m_k} \\ \Delta\theta_{m_k} \end{bmatrix} \tag{14a}$$

and

$$\begin{cases} \Delta\theta_i \approx \sin\Delta\theta = \dfrac{x_{si}\cos\theta_i - y_{si}\sin\theta_i}{\sqrt{(x_k - x_{si})^2 + (y_k - y_{si})^2}} \\ \Delta r_i = \sqrt{(x_k - x_{si})^2 + (y_k - y_{si})^2} - r_i \end{cases} \tag{14b}$$

in typical practical situations, (14) [44] are overly defined, so that a CLS solution was calculated from:

$$\boldsymbol{\Phi}\boldsymbol{\delta} = d \tag{15}$$

where,

$$\begin{aligned} \boldsymbol{\Phi} &= \boldsymbol{H}^T \boldsymbol{H} \\ &= \begin{bmatrix} \sum_i^{m_k} (\sin^2\theta_i + \dfrac{\cos^2\theta_i}{r_i^2}) & \sum_i^{m_k} (1 - \dfrac{1}{r_i^2}) \sin\theta_i \cos\theta_i \\ \sum_i^{m_k} (1 - \dfrac{1}{r_i^2}) \sin\theta_i \cos\theta_i & \sum_i^{m_k} (\cos^2\theta_i + \dfrac{\sin^2\theta_i}{r_i^2}) \end{bmatrix} \end{aligned} \tag{15a}$$

and

$$d = H^T \boldsymbol{\xi} = \begin{bmatrix} \sum_i^{m_k} (\Delta r_i \sin\theta_i + \dfrac{\Delta\theta_i \cos\theta_i}{r_i}) \\ \sum_i^{m_k} (\Delta r_i \cos\theta_i - \dfrac{\Delta\theta_i \sin\theta_i}{r_i}) \end{bmatrix} \tag{15b}$$

the solution of linear matrix (15) can be expressed in the form:

$$\boldsymbol{\delta} = (H^T H)^{-1}(H^T \boldsymbol{\xi}) = \boldsymbol{\Phi}^{-1} d \tag{16}$$

the vector $\boldsymbol{\xi}$ provides a correction estimate to modify the initial estimate of the two variables Δx and Δy. therefore, better estimates are:

$$\begin{bmatrix} x_{k|k} \\ y_{k|k} \end{bmatrix} = \begin{bmatrix} x_{k|k-1} \\ y_{k|k-1} \end{bmatrix} + \begin{bmatrix} \Delta x_k \\ \Delta y_k \end{bmatrix} \tag{17}$$

Equation (16) [44] is applied iteratively until the increments are sufficiently small. Note that these corrections are not the errors in the target predictive location, which are dependent on the measurement errors, but are increments in the iterative process. As the solution converges, these increments will approach zero in most situations, although the algorithm may not converge with large measurement errors.

D. NBC ANALYSIS

This part will give a brief review of NBC. Suppose that the decision attribute varies from $\{v_1, v_2, ..., v_c\}$, which implies that all effective measurement data x is categorized into c classes. The NBC is used to judge the membership degree of effective measurement data x belongs to v_i class (the target i). According to the prior probability and class-conditional probability of the remaining effective measurement data, Bayesian classifier calculates the posterior probability and determines the value of decision attribute for the remaining effective measurement data. The BC discriminates the class of effective measurement data x as following:

$$\begin{aligned} v(x) &= \underset{v_i, i=1,2,\cdots,c}{\arg\max} \{P(v_i|\tilde{x})\} \\ &= \underset{v_i, i=1,2,\cdots,c}{\arg\max} \left\{ \frac{P(v_i) P(\tilde{x}|v_i)}{P(x)} \right\} \\ &= \underset{v_i, i=1,2,\cdots,c}{\arg\max} \{P(v_i) P(\tilde{x}|v_i)\} \end{aligned} \tag{18}$$

where c is the number of classes and $P(v_i)$ is the prior probability of the i-th class, which can be estimated by the frequency of the effective measurement data of the i-th class, i.e., $P(v_i) = n_i/N$ in which N is the number of the remaining effective measurement data and n_i is the number of the effective measurement data in the i-th class. $P(x|v_i)$ is the class-conditional probability. The main purpose of naive BC is to estimate $P(x|v_i)$ according to the training effective measurement data in the i-th class. The class-conditional probability is as follows:

$$P(x|v_i) = P(x_1, x_2, \cdots, x_m|v_i) = \prod_{j=1}^{m} P\left(x_j|v_i\right) \tag{19}$$

Each effective measurement data is defined by m condition attributes, which are devoted to describe the specific characteristics of a measurement data. m condition attributes may contain information such as coordinates, distance from the predicted position, acceleration, density of measurement data, and sensor accuracy, and so on. Humans are taken as targets in this paper, and some conditional attributes can be set based on the related attributes of human normal walking. The velocity and acceleration of the target conform to the range of the walking for ordinary human. If the velocity or acceleration exceeds the threshold (the maximum velocity or acceleration of walking for ordinary human), $P(x|v_i) = 0$. Hence, using (19), the following decision rule of naive BC is obtained for determining the decision attribute value of the effective measurement data x.

$$v(x) = \underset{v_i, i=1,2,\cdots,c}{\arg\max} \left\{ \frac{n_i}{N} \prod_{j=1}^{m} P\left(x_j|v_i\right) \right\} \tag{20}$$

From (20), we can see that the calculation of $P(x|v_i)$ is the key to establish the association between the measurement data x and the class v_i by naive BC. According to the density estimation strategy, three methodologies NNB [45], FNB [46], and FNB$_{ROT}$ [47] are popular ways to estimate the component $P(x|v_i)$ for x.

1) NNB

Denote the elements in the i-th class as $x^{(i)}$. NNB [45] assumes that the $x^{(i)}$ obey a single Gaussian distribution. Then, $P(x|v_i)$ can be calculated from:

$$P(x_j|v_i) = \frac{1}{\sqrt{2\pi}\sigma_j^{(i)}} \exp\left[-\frac{\left(x_j - \mu_j^{(i)}\right)^2}{2\left(\sigma_j^{(i)}\right)^2} \right] \tag{21}$$

where $\mu_j^{(i)} = \frac{\sum_{l=1}^{n_i} x_l^{(i)}}{n_i}$ and $\left(\sigma_j^{(i)}\right)^2 = \frac{\sum_{l=1}^{n_i} \left[x_l^{(i)} - \mu_j^{(i)}\right]^2}{n_l}$ are the mean value and variance of all elements in $x^{(i)}$, respectively.

2) FNB

In many applications, to tackle the case of non-Gaussian distribution, John and Langley [46] proposed the FNB which estimates $P(x|v_i)$ through the following equation:

$$P(x_j|v_i) = \frac{1}{n_i h_j^{(i)}} \sum\nolimits_{l=1}^{n_i} \left[K\left(\frac{x_j - x_l^{(i)}}{h_j^{(i)}} \right) \right] \tag{22}$$

where $h_j^{(i)}$ is the bandwidth and $K(\cdot)$ is the kernel function. In FNB, $h_j^{(i)} = \frac{1}{\sqrt{n_i}}$ and $K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$. The experimental study shows that the classification performance of FNB mainly depends on the selection of the bandwidth $h_j^{(i)}$.

3) FNB$_{ROT}$

For evaluating the impact of different bandwidth parameter selection methods on the classification performance, Liu *et al.* [47] used the thumb rule to replace the traditional bandwidth parameter in FNB $h_j^{(i)} = \frac{1}{\sqrt{n_i}}$ with the following as:

$$h_j^{(i)} = \left(\frac{4}{3n_i} \right)^{\frac{1}{5}} \sigma_j^{(i)} \tag{23}$$

The kind of BC are called FNB$_{ROT}$. In addition to the rule of thumb mentioned earlier, we can also obtain other parameter selection methods from [48].

III. ANALYSIS OF IMPROVED WEIGHTED LEAST-SQUARES ALGORITHM

The measurements in WSNs are integrated to acquire more accurate position of the target. The novel outliers remove method, the ENNBC analysis [21] and WLF [49], [50] can be used to improve the performance of tracking system. In this section, the WLS is enhanced according to the ENNBC and novel outliers remove Method.

A. NOVEL OUTLIERS REMOVE AND DATASET DENSITY PEAKS FAST SEARCH METHOD

According to measurement dataset $X = \{x_1, \cdots, x_i, \cdots, x_N\}$ $(1 \le i \le N)$ from WSN, as shown in Fig. 3(a). ε neighborhood of x_i is expressed as the following equation:

$$N_\varepsilon(x_i) = \left\{ x_i, x_j \in X : d(x_j, x_i) \le \varepsilon \right\} \tag{24}$$

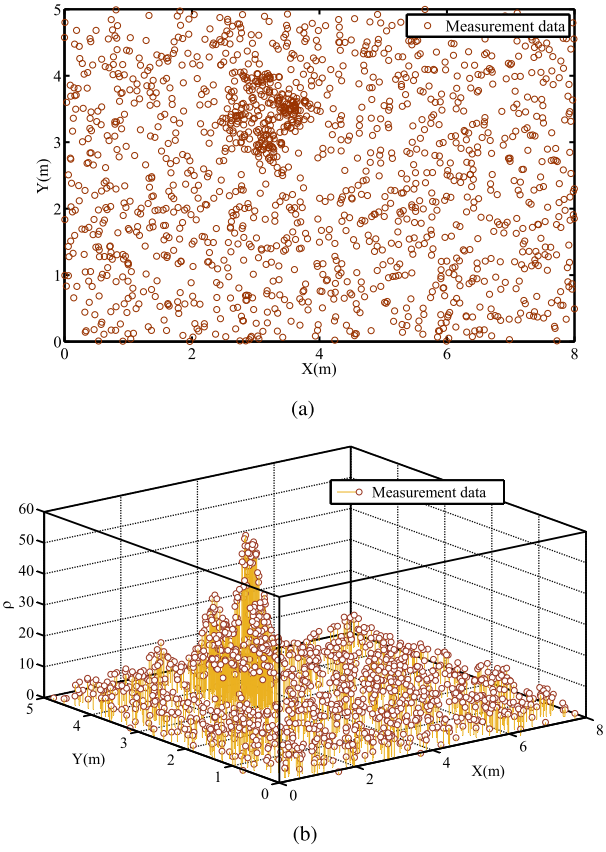


FIGURE 3. The measurement data from WSN: (a) Measurement data. (b) The 3D photograph of the location and density ρ of the measurement data.

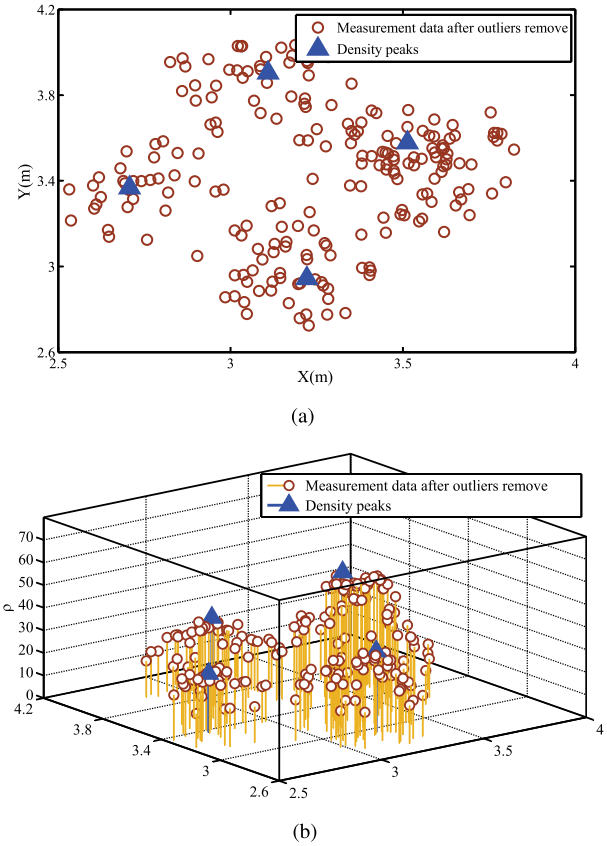


FIGURE 4. The remaining effective measurement data: (a) Remaining effective measurement data. (b) The 3D photograph of the location and density ρ of the remaining effective measurement data.

the local density $\rho_t(i)$ of measurement data x_i is expressed as the following equation:

$$\rho_t(x_i) = \sum_j \lambda(d(x_i, x_j) - \varepsilon) \tag{25}$$

where $\lambda(x) = 1$ if $x < 0$ and $\lambda(x) = 0$ otherwise. Basically, $\rho_k(x_i)$ is the number of measurement data that is closer than ε to measurement data x_i [35], as shown in Fig. 3(b). The algorithm is sensitive only to the relative magnitude of $\rho_k(x_i)$ in different data, implying that, for large data sets, the results of the analysis are robust with respect to the choice of ε. When $\rho_k(x_i) \leq \varepsilon_1$, x_i is the outlier, and m_k remaining effective measurement data is obtained, as shown in Fig. 4.

After the outliers is removed, since it is not known how many targets are included in the remaining measurements, the following calculations are needed to determine the number of classes (one class corresponds to one target)and the corresponding density peak. The method is based on the assumption that each class center is at a relatively large distance from any data with a higher local density, and they are surrounded by neighbors with lower local density. For every measurement data x_i, the local density $\rho_k(x_i)$ and the distance $\eta_k(i)$ from higher density data are calculated. Both these quantities depend only on the distances $d(x_i, x_j)$ between measurement data, which are assumed to satisfy the triangular inequality. $\eta_k(i)$ is calculated based on the minimum distance between the data x_i and any other measurement data with higher density, or the maximum distance between the data x_i and any other measurement data with lower density in ε_2 neighborhood of x_i, the expression is as follows:

$$\eta_k(i) = \begin{cases} \min\limits_{j:\rho_j > \rho_i} (d(x_i, x_j)), \\ \quad if\ \exists\ x_j,\ and\ \rho_k(x_j) > \rho_t(x_i), d(x_i, x_j) < \varepsilon_2 \\ \max\limits_{j:\rho_j < \rho_i} (d(x_i, x_j)),\ otherwise \end{cases} \tag{26}$$

for the data x_i, there is higher density in ε_2 neighborhood of x_i, we define $\eta_k(i) = \max_j(d(x_i, x_j))$, where x_j has the highest density. Note that $\eta_k(i)$ is much larger than the typical nearest neighbor distance only for data with maximum local density. In addition, the same class in measurement dataset may contain two or more peak density data. So it can be based on $\lambda(d(x_i, x_j) - \varepsilon_3)$ to determine whether a two density peak data belongs to the same class. If $\lambda(d(x_i, x_j) - \varepsilon_3) = 1$, the density peak data x_i and x_j are in same class. If there are multiple density peak data belonging to the same class, the class center is equal to the average value of these data. After the previous calculation, the measurement data Fig. 3(b) contains four classes, and the density peak position of each cluster is shown as the blue triangle in Fig. 4 The local

density value of each measurement data is calculated based on the position coordinate, and the density peak position of each class is also calculated. If the local density value of the measurement data is higher than the threshold, the data is considered to be effective measurement. Otherwise, it is outlier. Next, the effective measurement data with the highest local density value in a certain neighborhood around itself is searched, and considered to be the effective measurement data with the peak density value (be used to associate with known targets). However, the degree of association between each effective measurement data and the class needs further analysis (see following ENNBC algorithm analysis).

B. ENNBC ANALYSIS OF NEARBY TARGET MEASUREMENTS

In the above, the number of targets, effective measurement dataset $\tilde{X}$ after outliers remove, and density peaks were obtained. However, the relationship between the remaining effective measurement data $\tilde{x}$ and each target is unknown. After the analysis in Section III-A, the performance of the handling methodology is enhanced. Therefore, the method is redefined as ENNBC. Three methodologies NNB, FNB, and FNB$_{ROT}$ have the following two limitations: a) they are based on the assumption that all condition attributes are independent given the decision attribute, which is obviously not always valid in many practical applications. b) in the process of estimating the marginal probability density function of each attribute, ENNBC assumes that each attribute follows a Gaussian distribution problem; FNB/FNB$_{ROT}$ is suitable for the non-Gaussian distribution problem, while it has not an appropriate method of the parameter selection. The limitations seriously affect the accuracy of the probability density function estimation. In order to improve the classification performance based on removing or relaxing the above two limitations, the ENNBC method in which the restraint of independence among the attributes is removed and the joint probability density function estimation replaces the marginal probability density function estimations. ENNBC determines the class of the new effective measurement data $\tilde{x}$ as follows [25]:

$$\begin{aligned} \nu(\tilde{x}) &= \underset{\nu_i, i=1,2,\cdots,c}{\arg\max} \left\{ \frac{n_i}{N} P(\tilde{x}|\nu_i) \right\} \\ &= \underset{\nu_i, i=1,2,\cdots,c}{\arg\max} \left\{ \frac{1}{N h_i^m} \sum_{l=1}^{n_i} \left[K\left(\frac{x_1 - x_{l1}^{(i)}}{h_i}, \right.\right.\right. \\ &\qquad \left.\left.\left. \frac{x_2 - x_{l2}^{(i)}}{h_i}, \cdots, \frac{x_m - x_{lm}^{(i)}}{h_i} \right) \right] \right\} \end{aligned} \tag{27}$$

where $K(\cdot)$ is a multivariate kernel function and h_i is a crucial parameter called bandwidth. The multivariate Gaussian kernel is $K(\tilde{x}) = \frac{1}{\left(\sqrt{2\pi}\right)^m} \exp\left(-\frac{\tilde{x}\tilde{x}^T}{2}\right)$, where $\tilde{x}^T$ is the transpose of vector $\tilde{x}$. In addition, it is well acknowledged that the estimation performance of Parzen window method strongly relies on the selection of bandwidth h_i, and the detailed analysis for the optimal selection of h_i is given in [25, Sec. III-B]. Specifically, for a set of effective measurement data belonging to the i-th class, the optimal bandwidth $h_i (1 \leq i \leq c)$ can be simplified as:

$$h_i = \left(\frac{4m}{n_i |\sum_i|^{-\frac{1}{2}} \left(2tr\left(\sum_i^{-1} \sum_i^{-1} \right) + tr^2\left(\sum_i^{-1} \right) \right)} \right)^{\frac{1}{m+4}} \tag{28}$$

where $\sum_i = diag\left\{ \left(\sigma_1^{(i)}\right)^2, \left(\sigma_2^{(i)}\right)^2, \cdots, \left(\sigma_m^{(i)}\right)^2 \right\}$, and variance $\left(\sigma_j^{(i)}\right)^2, (1 \leq j \leq m)$ has been given in (21).

Next, we analyze the time complexities of the abovementioned four BC algorithms,i.e., NNB, FNB, FNB$_{ROT}$, and ENNBC. N denotes the number of training measurement data, M denotes the number of testing measurement data, and m denotes the number of condition attributes of the target. Since NNB needs to calculate the means and variances for the d condition attributes, the training time complexity of NNB is $T(Nm)$ and the classification time complexity is $T(Mm)$. FNB uses the superposition of N probability density functions of the Gaussian distribution to fit the true probability density function; thus, the training and classification time complexities of FNB are $T(Nm)$ and $T(MNm)$, respectively. FNB$_{ROT}$ used the rule of thumb to get some increase in the training time, however the training and classification time complexities remain $T(Nm)$ and $T(MNm)$, respectively [25]. Similar to FNB$_{ROT}$, the ENNBC also needs the additional time to remove outliers $T(R)$ and compute the optimal bandwidth in the training time. However, the elements in the dataset are greatly reduced after remove outliers, and the classification time complexity for the determination of the required parameter will decrease rather than increase. Therefore, the training

Algorithm 1 Outliers Remove

Require: dataset $X_k = \{x_{1,k}, \cdots, x_{i,k}, \cdots, x_{n,k}\} (1 \leq i \leq n)$.

Ensure: $\tilde{X}_k = \{\tilde{x}_{1,k}, \cdots, \tilde{x}_{i,k}, \cdots, \tilde{x}_{m_k,k}\} (1 \leq i \leq m_k)$, and centers C_k of classes.

1: **repeat**
2: the local density $\rho_t(i)$ of data $x_{i,k}$ was calculated by (25);
3: **until** the local densities of all measurements were calculated
4: **repeat**
5: using $\rho_t(i) \leq \varepsilon_1$ determining outliers;
6: **until** all outliers were found
7: **repeat**
8: the density peak was calculated by (26);
9: **until** all density peaks were calculated
10: **return**: results dataset $\tilde{X}_k = \{\tilde{x}_{1,k}, \cdots, \tilde{x}_{i,k}, \cdots, \tilde{x}_{m_k,k}\}$after outliers remove, and density peaks C_k of clusters in the dataset;

and classification time complexities of ENNBC are $T(R)$, $\leq T(Nm)$, and $\leq T(MNm)$ as well.

According to the ENNBC algorithm, the effective measurement data is associated with each class and assigned to the class with the greatest degree of association. However, the association between the classes and targets is unknown. The establishing method is to calculate the degree of association between known targets and the classes according to the Euclidean distance between the predicted positions and the effective measurement data with peak density (the predicted positions of targets can be calculated based on existing algorithms). After the measurement class is connected with the target, the optimal estimation of each target's at the current time is analyzed based on the following weighted least-squares algorithm.

C. WEIGHT OPTIMIZATION FOR THE WEIGHTED LEAST-SQUARES ALGORITHM

The measurement matrix H and the dependent variable $\boldsymbol{\xi}$ in (16) are processed by some filtering algorithms. However, the model parameters may contain negative factors in the solution for $\boldsymbol{\delta}$ by using (16), and the calculated result will be affected. These negative factors may be related to measurement noise or any other irrelevant factor. After the analysis of previous novel outliers remove and BC algorithms, the larger noise has been removed and effective measurements have been properly classified. After that, we need to use weighted Least-Squares algorithm to further analyze the effective measurement data for each target. According to previous BC algorithm, different weights are set for each effective measurement data, and the weight vector is $\hat{w}_k \in \mathbb{R}^{n\times 1}$.

$$P^{(i)} = \left\{\nu\left(\tilde{x}_1\right), \nu\left(\tilde{x}_2\right), \cdots, \nu\left(\tilde{x}_{n_i}\right)\right\} \tag{29}$$

using (29), the expected value of $\nu(\tilde{x})$ in i-th class is calculated from:

$$\mathbb{E}\left[\nu(\tilde{x}^{(i)})\right] = \mu_i \tag{30}$$

after the expected value of each class was calculated by using (30), the weight parameters of each effective measurement data are calculated by the bias parameter τ. The specific equation is as follows:

$$w_j^{(i)} = e^{\tau\left(\nu\left(\tilde{x}_j^{(i)}\right)-\mu_i\right)} \tag{31}$$

the weight parameters are normalized:

$$\hat{w}_j^{(i)} = \frac{w_j^{(i)}}{\sum_{l=1}^{n_i} w_l^{(i)}} \tag{32}$$

using (32), the weight vector $\hat{w}_j^{(i)}$ is calculated. Therefore, the diagonal weight matrix $\boldsymbol{\Lambda}$ is given by:

$$\Lambda = diag\left\{\hat{w}_1^{(i)}, \hat{w}_1^{(i)}, \hat{w}_2^{(i)}, \hat{w}_2^{(i)}, \cdots, \hat{w}_{n_i}^{(i)}, \hat{w}_{n_i}^{(i)}\right\} \tag{33}$$

a in (15), (16), and (32), the data matrix H is corrected by the diagonal weight matrix $\boldsymbol{\Lambda}$. The corrected equation is given as follows:

$$\hat{H} = \boldsymbol{\Lambda} H \tag{34}$$

the corresponding vector $\boldsymbol{\xi}$ is corrected by the diagonal weight matrix $\boldsymbol{\Lambda}$, and the corrected vector is as follows:

$$\hat{\boldsymbol{\xi}} = \left[\Delta r_1 \cdot \hat{w}_1^{(i)}, \Delta\theta_1 \cdot \hat{w}_1^{(i)}, \cdots, \Delta r_{n_i} \cdot \hat{w}_{n_i}^{(i)}, \Delta\theta_{n_i} \hat{w}_{n_i}^{(i)}\right]^T \tag{35}$$

the solution vector of the improved Bayesian enhanced least-squares algorithm can be obtained from:

$$\begin{aligned}\hat{\boldsymbol{\delta}} &= \left[(\boldsymbol{\Lambda}H)^T \boldsymbol{\Lambda}H\right]^{-1}\left[(\boldsymbol{\Lambda}H)^T \hat{\boldsymbol{\xi}}\right] \\ &= (\hat{H}^T\hat{H})^{-1}(\hat{H}^T\hat{\boldsymbol{\xi}}) = \hat{\boldsymbol{\Phi}}^{-1}\hat{d}\end{aligned} \tag{36}$$

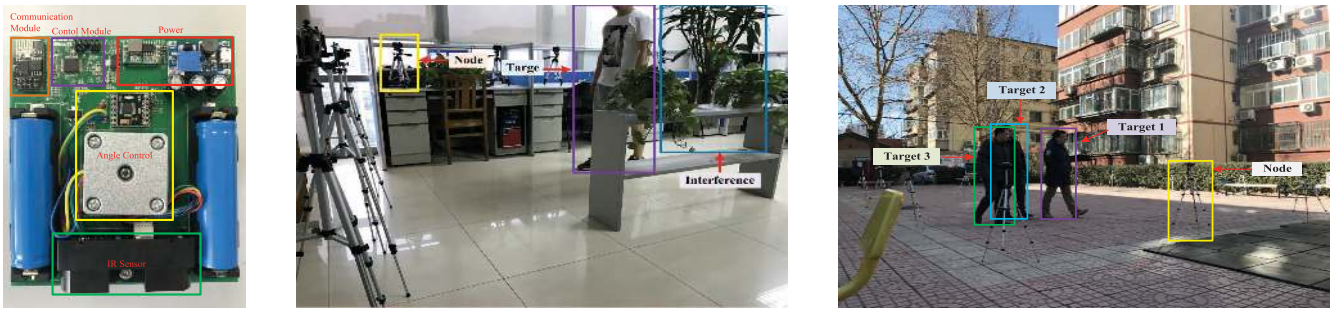
where $\hat{\boldsymbol{\Phi}}$ see equation (36a), as shown at the bottom of the next page, and

$$\hat{d} = \hat{H}^T\hat{\boldsymbol{\xi}} = \begin{bmatrix} \sum_{j=1}^{n_i}\left(\hat{w}_j^{(i)}\right)^2 \cdot \left(\Delta r_j \sin\theta_j + \dfrac{\Delta\theta_j \cdot \cos\theta_j}{r_j}\right) \\ \sum_{j=1}^{n_i}\left(\hat{w}_j^{(i)}\right)^2 \cdot \left(\Delta r_j \cos\theta_j + \dfrac{\Delta\theta_j \cdot \sin\theta_j}{r_j}\right) \end{bmatrix} \tag{36b}$$

using (36), the classical least-squares algorithm is corrected. First of all, in the analysis of novel outliers remove algorithm, the larger noise has been removed and the number of targets in the measurement of each cycle is obtained. Second, the remaining effective measurement data is accurately classified based on the BC algorithm, and the weighting parameter is calculated based on the probability of each effective measurement in i-th class. Finally, the measurement data of each cycle is fused according to the improved weighted least-squares algorithm. The details are presented in Algorithm 2.

Algorithm 2 Optimized Weighted Least-Squares Scheme

Require: Dataset $\tilde{X}_k = \left\{\tilde{x}_{1,k}, \cdots, \tilde{x}_{i,k}, \cdots, \tilde{x}_{n_k,k}\right\}$ $(1 \leq i \leq n_k)$, and centers $\boldsymbol{\theta}_t$ of clusters.
Ensure: Optimal location $(x_{k|k}, y_{k|k})$ of the target.
1: **repeat**
2: using ENNBC calculating $\nu\left(\tilde{x}_i\right)$ of effective measurement data $\tilde{x}_i$;
3: **until** $\nu\left(\tilde{x}\right)$ of every effective measurement data is calculated
4: using (29) calculating P_k;
5: using (30) calculating $\mathbb{E}\left[\nu\left(\tilde{x}^{(i)}\right)\right]$;
6: **repeat**
7: using (31) calculating $w_j^{(i)}$ of effective measurement data $\tilde{x}_j$;
8: **until** $w_j^{(i)}$ of every effective measurement data is calculated
9: Normalized $w_j^{(i)} \Rightarrow \hat{w}_j^{(i)}$;
10: Establish diagonal matrix $\hat{w}_j^{(i)} \Rightarrow \boldsymbol{\Lambda}$;
11: using (35) calculating $\hat{\boldsymbol{\xi}}$;
12: using (36) calculating $\hat{\boldsymbol{\delta}}$;
13: **return**: Optimal location $(x_{k|k}, y_{k|k})$ of the target.



(a) (b) (c)

FIGURE 5. The hardware experiment platform: (a) Sensor node. (b) Indoor scene 1. (c) Outdoor scene 2.

IV. HARDWARE EXPERIMENTAL PLATFORM AND SCENES

A. HARDWARE EXPERIMENTAL PLATFORM OF WSN

In order to verify the performance of the proposed algorithm, we designed a hardware experimental platform based on WSN. The experimental platform is remote and open tool for evaluating and comparing location and tracking algorithms. Each sensor node in the experimental platform contains ESP8266 WIFI from Espressif Systems as the communication module, the infrared ranging module for measured distance between the sensor and the target, and angle control for measured direction, as shown in Fig. 5(a). The whole experiment takes the human bodies as moving multi-target, and the indoor and outdoor scenes with interference are considered. In two scenes, we consider more than four sensor nodes to avoid blind areas caused by mutual occlusion between targets. More targets in both scenarios require more sensor nodes to eliminate blind zones. A total of 14 sensor nodes were deployed around the monitoring area in two scenes, as shown in Figs. 5(b) and 5(c). A series of experiments were carried out on the hardware experimental platform, and the performance of the proposed algorithm was evaluated. In the time test, each sampling period of all sensor nodes in WSN is less than $250 \times 10^{-3}s$. In the infrared ranging module and angle module, the relevant data acquisition takes about $200 \times 10^{-3}s$, the AD transforming and measurement data collection of sink node required about $30 \times 10^{-3}s$. When the target enters the measurement area of the sensor node, the activation mechanism required about $10 \times 10^{-3}s$.

B. EXPERIMENTAL SCENES

We first need to calibrate the position of each sensor node in two different scenarios, and plan an accurate measurement path to know the exact location of all moving targets at any time. The exact positions of these targets are used as real positions to verify the performance of the algorithms. In the experiment of the targets location and tracking, one or more targets move along metered path in the monitoring area at a normal speed of roughly $2m/s$, as shown in Fig. 5. The details of the indoor scene and the outdoor scene are as follows:

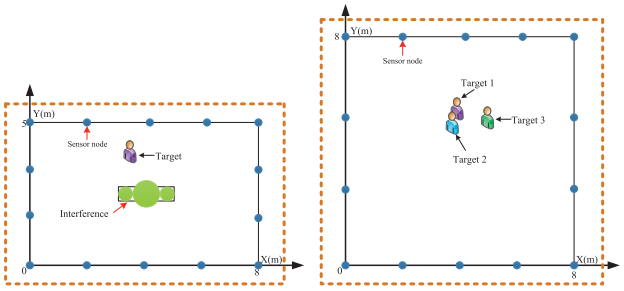
1) INDOOR SCENE 1

The experimental environment of the indoor scene is located in the room, as shown in Fig. 5(b) and Fig. 6(a). In this scene, we put some interference (a table and obstacles) inside the sensing area, 14 sensor nodes are located around a $5m \times 8m$ rectangle monitoring region. The position coordinates of the sensor nodes are SN1(0,0), SN2(2,0), SN3(4,0), SN4(6,0), SN5(8,0), SN6(8,1.667), SN7(8,3.333), SN8(8,5), SN9(6,5), SN10(4,5), SN11(2,5), SN12(0,5), SN11(0,3.333), and SN12(0,1.667) in meters, respectively. The height of every node is $1.15m$.

2) OUTDOR SCENE 2

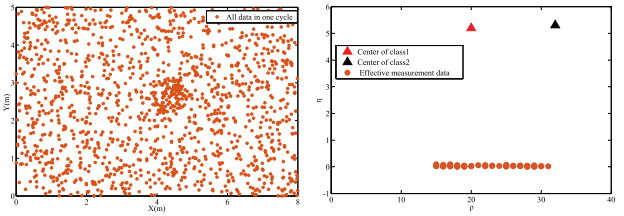
The experimental environment of the outdoor scene is located in open ground, as shown in Fig. 5(c) and Fig. 6(b). In this scene, three people walk around inside the sensing area, 14 sensor nodes are located around a $8m \times 8m$ square monitoring region. The position coordinates of the sensor nodes are SN1(0,0), SN2(2,0), SN3(4,0), SN4(6,0), SN5(8,0), SN6(8,2.667), SN7(8,5.333), SN8(8,8), SN9(6,8), SN10(4,8), SN11(2,8), SN12(0,8), SN11(0,5.333), and SN12(0,2.667) in meters, respectively. The height of every node is the same as above.

$$\hat{\mathbf{\Phi}} = \hat{H}^T \hat{H} = \begin{bmatrix} \sum_{j=1}^{n_i} \left(\hat{w}_j^{(i)}\right)^2 \cdot \left(\sin^2\theta_j + \frac{\cos^2\theta_j}{r_j^2}\right) & \sum_{j=1}^{n_i} (1 - \frac{1}{r_j^2}) \cdot \left(\hat{w}_j^{(i)}\right)^2 \cdot \sin\theta_j \cos\theta_j \\ \sum_{j=1}^{n_i} (1 - \frac{1}{r_j^2}) \cdot \left(\hat{w}_j^{(i)}\right)^2 \cdot \sin\theta_j \cos\theta_j & \sum_{j=1}^{n_i} \left(\hat{w}_j^{(i)}\right)^2 \cdot \left(\cos^2\theta_j + \frac{\sin^2\theta_j}{r_j^2}\right) \end{bmatrix} \tag{36a}$$



(a) (b)

FIGURE 6. The sketches for two scenes: (a) The sketches for indoor scene 1. (b) The sketches for outdoor scene 2.



(a) (b)

FIGURE 7. All measurement data in one cycle and decision graph of the Scene 1 in two-dimensions: (a) Distribution of 186 measurement points and 1314 noise points in one cycle. (b) Decision graph of the proposed classification algorithm based on all data points in (a).

V. PARAMETER DETERMINATION AND EXPERIMENT RESULTS

In this section, the proposed classification algorithm [51] is compared with the NNB, FNB, FNB_{ROT}, SVM, k-means, FCM, [52]–[54], PCM, and PFCM [55] on training datasets with respect to the four indexes, i.e., classification rate (CR) index, Dunn (D) index, CalinskiHarabasz (CH) index, and Silhouette (S) index. Then, the experimental results for single target and multi-target show that the proposed classification method can improve the performance of the range-based target tracking when it is used to classify measurement data. Meanwhile, the proposed classification method has better robustness against large localization and tracking errors.

A. CLASSIFICATION PERFORMANCE EVALUATION

The indoor scene 1 contains single target and interference (a table and obstacles) in Fig 5(b), and target as close as possible to intermediate interference. The sink node of sensor network receives the sample of 186 measurement points in one cycle and 1314 noise points with uniform distribution are increased. All these data points are set to training data and distributed in a two-dimensional coordinate system, as shown in Fig. 7(a). Obviously, we cannot judge the number of targets in simple ways. By analyzing the proposed classification algorithm, all data points are divided into noise and effective measurement of two targets due to the measurement error of sensor node and size of the human body, and the center c_1 and c_2 of two classes were obtained, as shown in Fig 8. In this experiment, we first consider the novel outliers remove and dataset density peaks fast search method. In Fig 8, effective measurement data and initial centers C_k of two classes are obtained, and the noise is brown points. Four neighborhood parameters ε, ε_1, ε_2, and ε_3 are $0.21m$, 15, $0.32m$, and $0.39m$, respectively. When the target localization and tracking system are certain, the optimal neighborhood parameters ε, ε_1, ε_2, and ε_3 of ENNBC can be applied to all remaining cycle. Next, we used the proposed classification algorithm to analyze the measurement data in Fig. 7(a). The number of classes equal to 2, and the initial centers of two classes are $c_1 = [4.0928, 2.3015]$ and $c_2 = [4.5281, 2.8640]$, respectively. The classes ν_1 and ν_2 contain 16 and 58 effective measurement data points, respectively. In ENNBC algorithm, data points with densities greater than the threshold are considered to be reliable measurement data points, which may result in low density data points being classified as noise. For each class, we look for the measurement data points with the highest density in their boundary regions. The data point whose density is greater than the density of all data points in its boundary region is considered to be the peak density (robust delegation) of this class. In addition, the ENNBC algorithm also consider the variance $(\sigma^{(i)})^2$ of each condition attribute of the measurement data. Thus, the classification is more accurate. Fig. 7(b) shows decision method of the proposed classification algorithm based on all data points in Fig. 7(a). Through this experiment, the ENNBC algorithm can successfully deal with data classification problems of adjacent classes of different densities.

Next, the measurement dataset $\tilde{x} = \{\tilde{x}^{(1)}, \tilde{x}^{(2)}\}$ of two classes, dataset $\tilde{x}^{(i)} = \left\{x_1^{(i)}, \cdots, x_j^{(i)}, \cdots, x_{n_i}^{(i)}\right\} (i = 1, 2), (1 \leq j \leq n_i)$ of i-th class(n_i is number of measurement data in i-th class), and the initial centers C_k of two classes are obtained according to above experiment. Then, we consider using the weighted Least-Squares algorithm to calculate the final data fusion centers. Finally, the data fusion centers of two classes are $c_1 = [4.1814, 2.3320]$ and $c_2 = [4.6162, 2.7846]$, respectively, as shown in Fig. 8.

Similarly, the Scene 2 contains three targets closely in Fig. 5(c), and three targets are as close as possible to each other. The sink node of sensor network received the sample of 211 measurement points in one cycle and 2089 noise points with uniform distribution are increased. All these data points are distributed in a two-dimensional coordinate system, as shown in Fig. 9(a). Obviously, we also can't judge the number of targets in simple ways. After analysis of the proposed classification algorithm, all data points are divided into noise and effective measurement of three targets, and the center $c_1 = [5.2185, 5.0680]$, $c_2 = [4.6713, 4.3100]$, and $c_3 = [5.5543, 5.5058]$ of three classes were obtained, as shown in Fig. 10. The classes ν_1, ν_2, and ν_3 contain 67, 50, and 49 effective measurement data points, respectively. Fig. 9(b) shows decision method of the proposed classification algorithm based on all data points in Fig. 9(a). The measurement dataset $\tilde{x} = \{\tilde{x}^{(1)}, \tilde{x}^{(2)}, \tilde{x}^{(3)}\}$ of three classes,

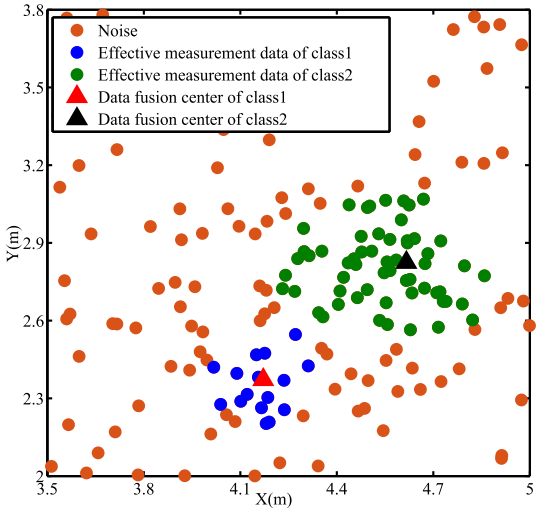


FIGURE 8. Final results of proposed classification algorithm for the data in Fig. 7(a).

dataset $\tilde{x}^{(i)} = \left\{x_1^{(i)}, \cdots, x_j^{(i)}, \cdots, x_{n_i}^{(i)}\right\} (i = 1, 2, 3), (1 \leq j \leq n_i)$ of i-th class, and the initial centers C_k of three classes are obtained according to this experiment. Then, we also use the weighted Least-Squares algorithm to calculate the final data fusion centers. Finally, the data fusion centers of three classes are $c_1 = [5.2853, 5.1409]$, $c_2 = [4.7325, 4.3932]$, and $c_3 = [5.5730, 5.5494]$, respectively, as shown in Fig. 10.

In the following, the performance of the proposed classification method is compared with that of the NNB, FNB, FNB_{ROT}, SVM, k-means, FCM, PCM, and PFCM algorithms on training datasets. we use the four indexes, i.e., D index, CH index, S index, and CR index, to compare a class with the real data label information. These four indexes can be described as:

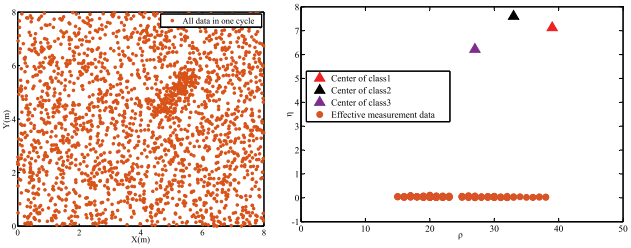
1) D INDEX

$$D(v) = \min\left\{\min \frac{\min\limits_{x_i^{(l)} \in v_l, x_j^{(l)} \in v_l} d\left(x_i^{(l)}, x_j^{(l)}\right)}{\max\left\{\max\limits_{x_i^{(l)} \in v_l, x_j^{(l)} \in v_l} d\left(x_i^{(l)}, x_j^{(l)}\right)\right\}}\right\} \quad (37)$$

where $x_i^{(l)}$ is i-th effective measurement data point in l-th class. D index was recommended for recognition compact and well separated classes. After simple analysis (refequ:dmyy), it can be concluded that the greater D is, the better classification performance is [13].

2) CH INDEX

$$CH(v) = \frac{\frac{1}{c-1}\sum_i^c n_i d^2(c_i, c_0)}{\frac{1}{N-c}\sum_{i=1}^c \sum_{j=1, x_j^{(i)} \in v_i}^{n_i} d^2\left(x_j^{(i)}, c_i\right)} \quad (38)$$



(a) (b)

FIGURE 9. All measurement data in one cycle and decision graph of the Scene 2 in two-dimensions: (a) Distribution of 211 measurement points in one cycle and we increased 2089 noise points in one cycle. (b) Decision graph of the proposed classification algorithm based on all data points in (a).

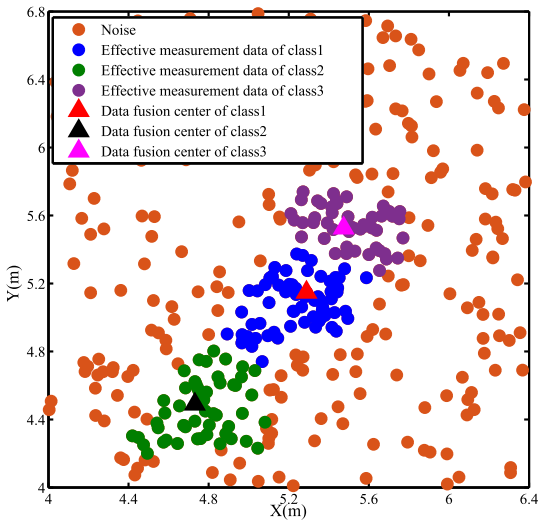


FIGURE 10. Final results of proposed classification algorithm for the 2300 data points in Fig. 9(a).

where c_0 is the center of the dataset and n_c is the number of all data. CH index is calculated for the ratio of separation and affinity degree. Therefore, it can be concluded that the greater CH is, the closer between data in same class and more scattered between classes [13].

3) S INDEX

$$S(v) = \frac{1}{c}\sum_{i=1}^c \left\{\frac{1}{n_i}\sum_{j=1, x_j^{(i)} \in v_i}^{n_i} \frac{b\left(x_j^{(i)}\right) - a\left(x_j^{(i)}\right)}{\max\left[b\left(x_j^{(i)}\right), a\left(x_j^{(i)}\right)\right]}\right\} \quad (39)$$

where,

$$a\left(x_j^{(i)}\right) = \frac{1}{n_i - 1}\sum_{l=1, x_j^{(i)}, x_l^{(i)} \in v_i, x_j^{(i)} \neq x_l^{(i)}}^{n_i} d\left(x_j^{(i)}, x_l^{(i)}\right) \quad (40)$$

$$b\left(x_j^{(i)}\right) = \min_{l, l \neq i}\left\{\frac{1}{n_l}\sum_{x_j^{(i)} \in v_i, x_k^{(l)} \in v_l} d\left(x_j^{(i)}, x_k^{(l)}\right)\right\} \quad (41)$$

S index is used to measure the classification performance by calculating the distance between two elements from different classes and the distance between two elements in the same class. Similar to the above, the greater the value of S is, the better the classification performance of the algorithm is.

4) CR

CR measures the percentage of the points that have been correctly labeled by each algorithm. The CR is determined first by transforming the fuzzy partition matrix into a Boolean partition matrix and by selecting the cluster with the maximum membership value for each pattern. Class labels are assigned to each class based on the class that dominates that class. the greater the value of CR is, the better the classification performance of the algorithm is. For these algorithms, the CR is often used as an important index to evaluate the performance [56].

The performance of the NNB, FNB, FNB$_{ROT}$, SVM, k-means, FCM, PCM, PFCM, and ENNBC is showed in Table 2 based on the Scene 1, respectively. As is well-known, when the NNB, FNB, FNB$_{ROT}$, k-means, and FCM are initialized with an exact number of classes. They are unable to resolve underlying classification structures, which is caused by the noise in the dataset and the great difference of variances between adjacent classes. Finally, Table 2 shows that the ENNBC algorithm has the best performance based on D, CH, S, and CR four indexes, calculating more accurately the real centers of the classes, and it requires the least iterations for convergence. It is worth noting that the operation time of ENNBC algorithm is less than that of the FNB, FNB$_{ROT}$, SVM, and PFCM, which can satisfy the real-time requirement of target tracking. This is due to the ENNBC algorithm runs novel outliers remove and dataset density peaks fast search method first, a large number of outliers in dataset have been removed which greatly reduces iterations of the algorithm, and the initial center C_k of the class is very close to the real value. In addition, the performance of the NNB, FNB, FNB$_{ROT}$, SVM, k-means, FCM, PCM, PFCM, and ENNBC is showed in Table 3 based on the Scene 2, respectively. Similarly, the ENNBC algorithm has the best performance based on these four indexes, calculating more accurately the real centers of the classes, and it can also satisfy real-time requirement of multi-target tracking.

TABLE 2. Performance of classification algorithms in the scene 1.

Algorithm	D	CH	S	CR	Time(s)
NNB	0.0061	182.42	0.2814	39.64	0.0972
FNB	0.0084	201.71	0.3104	45.59	0.2213
FNB$_{ROT}$	0.0101	214.54	0.3278	47.38	0.2818
SVM	0.0018	52.82	0.1832	32.20	5.4362
k-means	0.0079	60.52	0.2037	22.69	0.0910
FCM	0.0085	139.44	0.2330	24.92	0.1162
PCM	0.0024	58.30	0.2043	52.34	0.1223
PFCM	0.0351	731.22	0.7216	83.71	0.1904
ENNBC	**0.0382**	**866.41**	**0.7601**	**87.89**	**0.1592**

TABLE 3. Performance of classification algorithms in the scene 2.

Algorithm	D	CH	S	CR	Time(s)
NNB	0.0055	168.80	0.2759	30.60	0.2174
FNB	0.0071	189.53	0.3194	39.88	0.4419
FNB$_{ROT}$	0.0098	208.77	0.3360	42.31	0.5714
SVM	0.0021	38.76	0.1949	23.67	15.8120
k-means	0.0081	58.82	0.2190	21.01	0.2093
FCM	0.0094	129.75	0.2259	23.38	0.2650
PCM	0.0016	48.37	0.1947	50.59	0.2697
PFCM	0.0368	724.08	0.7135	81.93	0.3522
ENNBC	**0.0409**	**786.97**	**0.7855**	**88.25**	**0.2831**

B. TRACKING PERFORMANCE EVALUATION FOR SINGLE TARGET

We first design a scenario like this: a moving target enters the monitoring area of the sensor network. the target is close to the disturbance and walks around it in the indoor Scene 1. Fig. 11 shows the target tracking results of two different algorithms in the indoor Scene 1. The real moving trajectory on the ground is marked as dotted line, and the trajectories of the algorithms calculation are marked as solid line, red and blue lines correspond to proposed algorithm and EKF algorithm, respectively. The EKF algorithm in this section used for 2-D maneuvering target tracking is based on the EKF-CMAC algorithm in [44]. In the previous section, we discussed that interferents are also measured, but they are immovable in each period, so the final fusion position of the interference in each period is basically the same, and we defined it as an immovable target, which is not considered in this analysis. In Fig. 11, the calculated trajectories by two algorithms are compared (the standard error of the sensor node localization is about 20cm). The tracking results show that the accuracy of estimated trajectory based on the proposed algorithm in the paper are better than those based on EKF algorithm. Fig. 12 shows the position error (PE) between the estimated location and the real ground location of the target at each

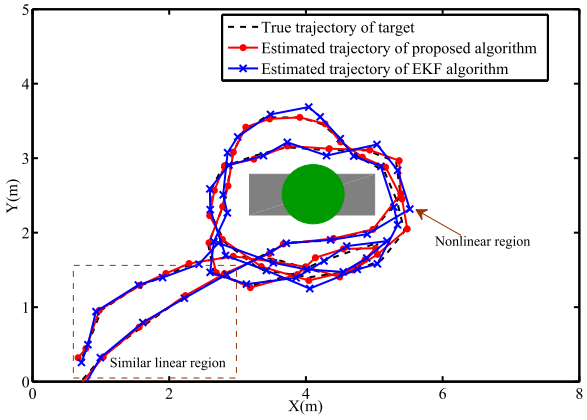


FIGURE 11. The tracking results of single target based on EKF and proposed algorithm in Scene 1.

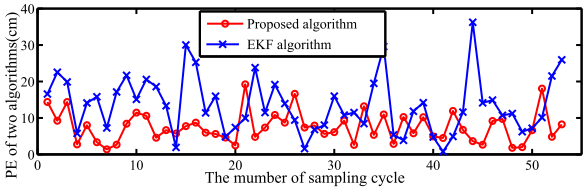


FIGURE 12. Comparison of two different algorithms for the PE.

TABLE 4. Comparison of proposed algorithm and EKF algorithm for RMSE.

Target	RMSE(cm)		Improvement rate (%)
	Proposed Method	EKF	
1	8.3081	11.0529	24.83

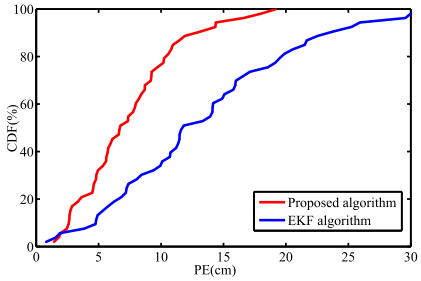


FIGURE 13. CDFs of tracking PE.

TABLE 5. Comparison of RMSE for proposed algorithm and EKF based on three targets.

Target	RMSE(cm)		Improvement rate (%)
	Proposed Method	EKF	
1	8.9056	11.8932	25.12
2	8.6618	11.4794	24.54
3	8.4041	11.0870	24.20

cycle according to the proposed algorithm and EKF. The PE based on proposed algorithm is smaller for comparing tracking results. The root-mean-square errors (RMSEs) of proposed algorithm and EKF are shown in Table 4, and the RMSE of the target location has been reduced by no less than 24 percent. Fig. 13 shows the cumulative distribution functions (CDFs) of proposed algorithm and EKF for the tracking PE in indoor scene 1. We can easily observe that the tracking PE of the proposed algorithm is less than EKF algorithm. After calculation and analysis, 94 percent of the PE based on proposed algorithm are less than 12.28*cm* in all cycle, while 94 percent of the PE based on EKF are less than 22.49*cm*, a 45.4 percent improvement.

C. TRACKING PERFORMANCE EVALUATION FOR MULTI-TARGET TRACKING

Similarly, we first design a scenario like this: Three nearby moving targets enter simultaneously monitoring area of the sensor network in the outdoor Scene 2. The tracking results of three moving targets based on two different algorithms are

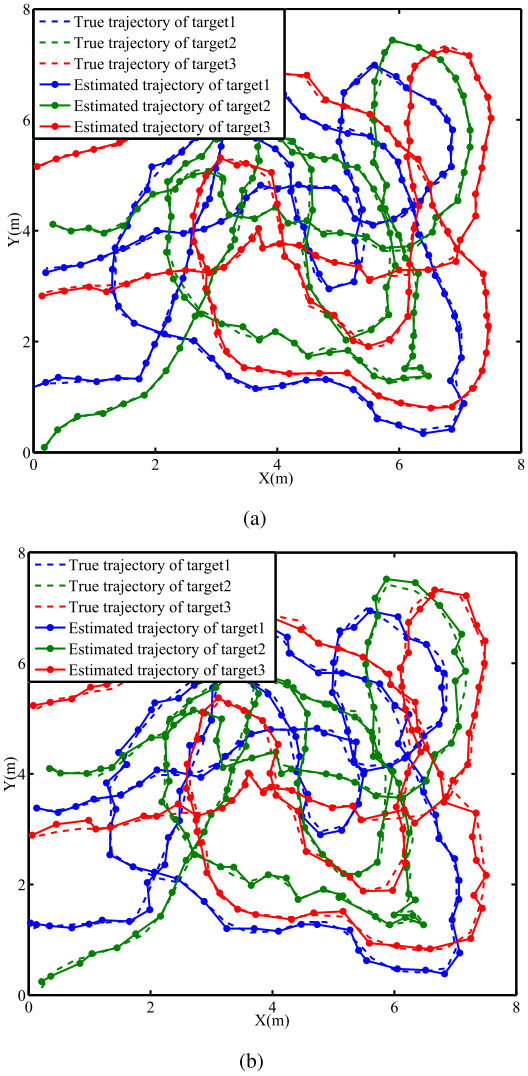


FIGURE 14. The tracking results of three targets in outdoor scene 2 based on two algorithms: (a) Proposed algorithm. (b) EKF.

shown in Fig. 14. The real moving trajectories on the ground are marked as dotted lines, and the trajectories of the algorithms calculation are marked as solid lines. Different targets correspond to different colors. In Fig. 14, the calculated trajectories by two algorithms are compared (the standard error of the sensor node localization is about 20*cm*). The same as the previous part, the tracking results show that the accuracy of estimated trajectories based on the proposed algorithm in the paper are better than those based on EKF algorithm. Fig. 15 shows the PE between the estimated location and the real ground location of three moving targets at each cycle according to the proposed algorithm and EKF. The PE based on proposed algorithm is smaller for comparing tracking results. Comparison of RMSE for proposed algorithm and EKF based on three moving targets is shown in Table 5. The RMSE of the target location has been reduced by no less than 24 percent. Fig. 16 shows the CDFs of proposed algorithm and EKF for the tracking PE in the outdoor scene 2. We can also easily observe that the tracking PE of the proposed

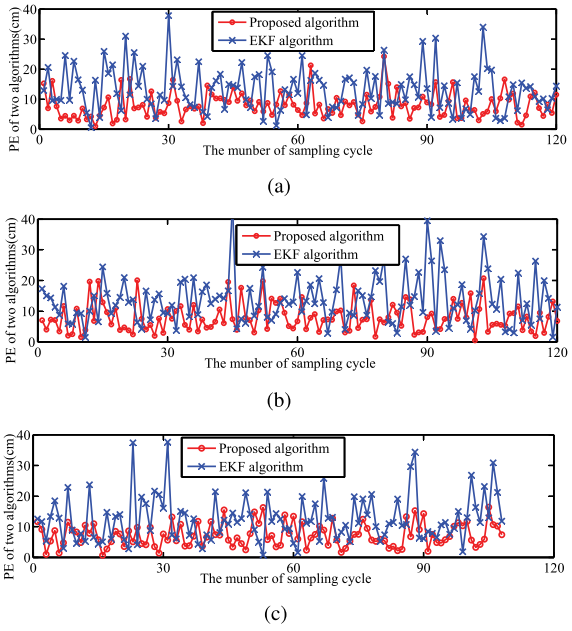


FIGURE 15. Comparison of the PE for three targets based on two algorithms: (a) The PE of target 1. (b) The PE of target 2. (c) The PE of target 3.

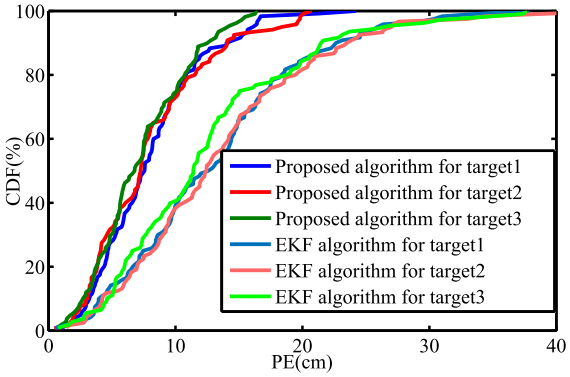


FIGURE 16. CDFs of tracking PE.

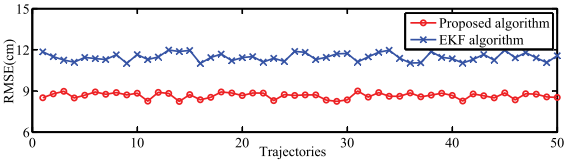


FIGURE 17. Comparison of the RMSE for two algorithms based on 50 trajectories.

algorithm is less than EKF algorithm. In addition, we calculate the RMSEs of two algorithms based on 50 trajectories for better illustrate the performance of the proposed algorithm, as shown in Fig. 17. In conclusion, the proposed algorithm improves accuracy of range-based multi-target tracking in compared with EKF algorithm.

VI. CONCLUSION

The main works of the paper are concluded as follows: 1) the measurement data density estimation is successfully applied to effective measurement data identification and classification, which effectively improves the traditional direct classification method and takes the dependence among continuous density attributes into account; 2) The weight values of each effective measurement data are optimized by using the density and probability information calculated in the previous steps, and the optimized weights are applied to the weighted least-squares algorithm; and 3) The detailed experimental results show that the proposed algorithm based on four indicators has the best classification performance, which can calculate the real centers of the classes more accurately and meet the real-time requirements of multi-target tracking. Meanwhile, the single target and multi-target tracking experiments show the provided enhancement of the tracking performance when proposed algorithm is used measurement data classification. The RMSE based on proposed algorithm has been reduced by no less than 24 percent in compared with EKF. Our future research content is to introduce image factors into existing multi-target localization and tracking, and improve and optimize the corresponding image recognition algorithms.

REFERENCES

[1] H. Fourati, *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications*. Boca Raton, FL, USA: CRC Press, 2016.
[2] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 19, no. 1, pp. 5–18, Jan. 2004.
[3] Z. Sunberg, S. Chakravorty, and R. S. Erwin, "Information space receding horizon control for multisensor tasking problems," *IEEE Trans. Cybern.*, vol. 46, no. 6, pp. 1325–1336, Jun. 2016.
[4] Z. Yan, X. Liu, J. Zhou, and D. Wu, "Coordinated target tracking strategy for multiple unmanned underwater vehicles with time delays," *IEEE Access*, vol. 6, pp. 10348–10357, 2018.
[5] J. Luo, Z. Zhang, C. Liu, and H. Luo, "Reliable and cooperative target tracking based on WSN and WiFi in indoor wireless networks," *IEEE Access*, vol. 6, pp. 24846–24855, 2018.
[6] T. Wang, R. C. De Lamare, and A. Schmeink, "Alternating optimization algorithms for power adjustment and receive filter design in multihop wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 173–184, Jan. 2015.
[7] X. Yu and J. Liang, "Genetic fuzzy tree based node moving strategy of target tracking in multimodal wireless sensor network," *IEEE Access*, vol. 6, pp. 25764–25772, 2018.
[8] R. Choudhuri and R. K. Das, "Coverage of targets in mobile sensor networks with restricted mobility," *IEEE Access*, vol. 6, pp. 10803–10813, 2018.
[9] E. Kim and K. Kim, "Distance estimation with weighted least squares for mobile beacon-based localization in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 17, no. 6, pp. 559–562, Jun. 2016.
[10] A. Khan, B. Rinner, and A. Cavallaro, "Cooperative robots to observe moving targets," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 187–198, Jan. 2018.
[11] D. Tick, A. C. Satici, J. Shen, and N. Gans, "Tracking control of mobile robots localized via chained fusion of discrete and continuous epipolar geometry, IMU and Odometry," *IEEE Trans. Cybern.*, vol. 43, no. 4, pp. 1237–1250, Aug. 2013.
[12] S. Bhattacharya, H. Blunck, M. B. Kjærgaard, and P. Nurmi, "Robust and energy-efficient trajectory tracking for mobile devices," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 430–443, Feb. 2014.
[13] T. Wang, W. Xiang, Z. Zhao, Z. He, and T. Xia, "Measurement data classification optimization based on a novel evolutionary kernel clustering algorithm for multi-target tracking," *IEEE Sensors J.*, vol. 18, no. 9, pp. 3722–3733, Mar. 2018.
[14] I. Sharp, K. Yu, and M. Hedley, "On the GDOP and accuracy for indoor positioning," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2032–2051, JUL. 2012.

[15] M. K. Kalandros, L. Trailovic, L. Y. Pao, and Y. Bar-Shalom, "Tutorial on multisensor management and fusion algorithms for target tracking," in *Proc. Amer. Control Conf.*, vol. 5, Jul. 2004, pp. 4734–4748.

[16] Y. Wu, Y. Sui, and G. Wang, "Vision-based real-time aerial object localization and tracking for UAV sensing system," *IEEE Access*, vol. 5, pp. 23969–23978, 2017.

[17] L. Lin and M. Zhu, "Efficient tracking of moving target based on an improved fast differential evolution algorithm," *IEEE Access*, vol. 6, pp. 6820–6828, 2018.

[18] X. Wang, M. Fu, and H. Zhang, "Target tracking in wireless sensor networks based on the combination of KF and MLE using distance measurements," *IEEE Trans. Mobile Comput.*, vol. 11, no. 4, pp. 567–576, Apr. 2012.

[19] A. Nakib, B. Daachi, M. Dakkak, and P. Siarry, "Mobile tracking based on fractional integration," *IEEE Trans. Mobile Comput.*, vol. 13, no. 10, pp. 2306–2319, Oct. 2014.

[20] S. Zhang, S. Zhao, S. Yao, and Z. Li, "Single object tracking with fuzzy least squares support vector machine," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5723–5738, Dec. 2015.

[21] L. Jiang, S. S. Singh, and S. Yıldırım, "Bayesian tracking and parameter learning for non-linear multiple target tracking models," *IEEE Trans. Signal Process.*, vol. 63, no. 21, pp. 5733–5745, Nov. 2015.

[22] X. Wang *et al.*, "Track fusion based on threshold factor classification algorithm in wireless sensor networks," *Int. J. Commun. Syst.*, vol. 30, no. 7, p. e3164, 2016.

[23] B. Fortin, S. Hachour, and F. Delmotte, "Multi-target PHD tracking and classification using imprecise likelihoods," *Int. J. Approx. Reasoning*, vol. 90, pp. 17–36, Nov. 2017.

[24] S. Hachour, F. Delmotte, D. Mercier, and E. Lefèvre, "Object tracking and credal classification with kinematic data in a multi-target context," *Inf. Fusion*, vol. 20, no. 1, pp. 174–188, 2014.

[25] X.-Z. Wang, Y.-L. He, and D. D. Wang, "Non-Naïve Bayesian classifiers for classification problems with continuous attributes," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 21–39, Jan. 2014.

[26] H. H. Kim, K. N. Ha, S. Lee, and K. C. Lee, "Resident location-recognition algorithm using a Bayesian classifier in the PIR sensor-based indoor location-aware system," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 2, pp. 240–245, Mar. 2009.

[27] S. D. Xenaki, K. D. Koutroumbas, and A. A. Rontogiannis, "A novel adaptive possibilistic clustering algorithm," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 4, pp. 791–810, Aug. 2015.

[28] Z. Zhao, X. Wang, and T. Wang, "A novel measurement data classification algorithm based on SVM for tracking closely spaced targets," *IEEE Trans. Instrum. Meas.*, to be published, doi: 10.1109/TIM.2018.2861107.

[29] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5933–5942, Dec. 2016.

[30] W. Bi, M. Cai, M. Liu, and G. Li, "A big data clustering algorithm for mitigating the risk of customer churn," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1270–1281, Jun. 2016.

[31] Y. Wu, H. Shen, and Q. Z. Sheng, "A cloud-friendly RFID trajectory clustering algorithm in uncertain environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 8, pp. 2075–2088, Aug. 2015.

[32] J. Zhou, C. L. P. Chen, L. Chen, and H. X. Li, "A collaborative fuzzy clustering algorithm in distributed network environments," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 6, pp. 1443–1456, Dec. 2014.

[33] B.-G. Hu, "What are the differences between Bayesian classifiers and mutual-information classifiers?" *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 249–264, Feb. 2014.

[34] D. Fisch, E. Kalkowski, and B. Sick, "Knowledge fusion for probabilistic generative classifiers with data mining applications," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 652–666, Mar. 2014.

[35] A. Rodriguez and A. Laio, "Machine learning. Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[36] G. Gan and M. K.-P. Ng, "K-means clustering with outlier removal," *Pattern Recognit. Lett.*, vol. 90, pp. 8–14, Apr. 2017.

[37] F. Pernkopf, M. Wohlmayr, and S. Tschiatschek, "Maximum margin Bayesian network classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 521–532, Mar. 2012.

[38] M. Nazari, J. Shanbehzadeh, and A. Sarrafzadeh, "Fuzzy C-means based on automated variable feature weighting," in *Proc. Int. MultiConf. Eng. Comput. Sci.*, 2013, pp. 25–29.

[39] B. A. Pimentel and R. M. C. R. de Souza, "A weighted multivariate fuzzy C-means method in interval-valued scientific production data," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3223–3236, 2014.

[40] L. Zhang, W. Pedrycz, W. Lu, X. Liu, and L. Zhang, "An interval weighed fuzzy C-means clustering by genetically guided alternating optimization," *Expert Syst. Appl.*, vol. 41, no. 13, pp. 5960–5971, 2014.

[41] N. Enayati, E. D. Momi, and G. Ferrigno, "A quaternion-based unscented Kalman filter for robust optical/inertial motion tracking in computer-assisted surgery," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 8, pp. 2291–2301, Aug. 2015.

[42] Y. Yu, "Consensus-based distributed mixture Kalman filter for maneuvering target tracking in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8669–8681, Oct. 2016.

[43] J. Chen, J. Li, S. Yang, and F. Deng, "Weighted optimization-based distributed Kalman filter for nonlinear target tracking in collaborative sensor networks," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3892–3905, Nov. 2017.

[44] C.-M. Lin and C.-S. Hsueh, "Adaptive EKF-CMAC-based multisensor data fusion for maneuvering target," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 7, pp. 2058–2066, Jul. 2013.

[45] R. R. Bouckaert, "Naive bayes classifiers that perform well with continuous variables," in *Proc. Ai Adv. Artif. Intell., 17th Austral. Joint Conf. Artif. Intell.*, Cairns, Australia: DBLP, Dec. 2004.

[46] G. H. John and P. Langley, "Estimating continuous distributions in Bayesian classifiers," in *Proc. 11th Conf. Uncertainty Artif. Intell.*, 1995, pp. 338–345.

[47] M. Liu, X. Jiang, and A. C. Kot, "A multi-prototype clustering algorithm," *Pattern Recognit.*, vol. 42, no. 5, pp. 689–698, 2009.

[48] A. Quinteladelrio and G. Estevezperez, "Nonparametric kernel distribution function estimation with kerdiest: An R package for bandwidth choice and applications," *J. Stat. Softw.*, vol. 50, no. 8, pp. 1–21, 2012.

[49] Y. Chi, Y. C. Eldar, and R. Calderbank, "PETRELS: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5947–5959, Dec. 2013.

[50] M. K. Awad and K. T. Wong, "Recursive least-squares source tracking using one acoustic vector sensor," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3073–3083, Oct. 2012.

[51] C. L. Chowdhary and D. P. Acharjya, "A hybrid scheme for breast cancer detection using intuitionistic fuzzy rough set technique," *Int. J. Healthcare Inf. Syst. Inform.*, vol. 11, no. 2, pp. 1–24, 2016.

[52] C. L. Chowdhary and D. P. Acharjya, "Breast cancer detection using intuitionistic fuzzy histogram hyperbolization and possibilitic fuzzy C-mean clustering algorithms with texture feature based classification on mammography images," in *Proc. Int. Conf. Adv. Inf. Commun. Technol. Comput.*, 2016, p. 21.

[53] C. L. Chowdhary and D. P. Acharjya, "Segmentation of mammograms using a novel intuitionistic possibilistic fuzzy c-mean clustering algorithm," in *Proc. 50th Annu. Conv. Comput. Soc. India, Nature Inspired Comput., Adv. Intell. Syst. Comput.*, vol. 652. CSI, Dec. 2015, pp. 75-82.

[54] C. L. Chowdhary and D. P. Acharjya, "Clustering algorithm in possibilistic exponential fuzzy C-mean segmenting medical images," *J. Biomimetics, Biomater. Biomed. Eng.*, vol. 30, pp. 12–23, Jan. 2017.

[55] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, "A possibilistic fuzzy C-means clustering algorithm," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 4, pp. 517–530, Aug. 2005.

[56] T. Chaira, "A novel intuitionistic fuzzy C means clustering algorithm and its application to medical images," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1711–1717, 2011.

XINXIN HE received the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2017, where she is currently with the School of Information and Communication Engineering. Her research interests include VANET, cognitive radio networks, and target localization and tracking.

TAO WANG was born in Qianjin, Heyang, Shaanxi, in 1986. He is currently pursuing the Ph.D. degree in microelectronics and solid electronics with Beihang University, Beijing, China, where he has been with the School of Electronic and Information Engineering, since 2014. His current research interests include data fusion, target localization and tracking, and space-sky information networks.

WEI LIU was born in Dezhou, Shandong. He is currently pursuing the Ph.D. degree in communication and information systems with the Beijing University of Posts and Telecommunications. His research interests include visible light communication, the Internet of Vehicles, and target localization and tracking.

TAO LUO (M'09–SM'15) received the B.S. degree from Shaanxi Normal University, in 1993, the M.S. degree from Shanghai University, in 1999, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, China, in 2002, where he is currently a Professor. His research interests include mobile communication, cognitive radio networks, the Internet of Vehicles, machine learning, and privacy preservation.