

Research Article

An Online Multisensor Data Fusion Framework for Radar Emitter Classification

Dongqing Zhou, Xing Wang, Siyi Cheng, and Xi Zhang

Aeronautics and Astronautics Engineering College, Air Force Engineering University, Xi'an, Shaanxi 710038, China

Correspondence should be addressed to Dongqing Zhou; qq-eastz@126.com

Received 1 October 2015; Accepted 3 April 2016

Academic Editor: Hikmat Asadov

Copyright © 2016 Dongqing Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Radar emitter classification is a special application of data clustering for classifying unknown radar emitters in airborne electronic support system. In this paper, a novel online multisensor data fusion framework is proposed for radar emitter classification under the background of network centric warfare. The framework is composed of local processing and multisensor fusion processing, from which the rough and precise classification results are obtained, respectively. What is more, the proposed algorithm does not need prior knowledge and training process; it can dynamically update the number of the clusters and the cluster centers when new pulses arrive. At last, the experimental results show that the proposed framework is an efficacious way to solve radar emitter classification problem in networked warfare.

1. Introduction

Radar emitter classification is an important subject in electromagnetism surveillance [1–3], which is widely used in the airborne electronic support (ES) system [4]. It is a clustering problem which is used to process the sampled pulses radiated from unknown radars. This subject focuses on two main tasks [5]: (1) to confirm the number of radar emitters and (2) to classify which emitter the incoming pulses belong to. The radar emitter classification uses the pulse parameters, such as pulse repetition interval (PRI), direction-of-arrival (DOA), pulse frequency (PF), pulse width (PW), pulse amplitude (PA), and intrinsic pulse shape, to cluster the received radar pulses. In applications, the radar emitter classification can be used in electromagnetism environment monitoring and special radar emitter identification, especially those from hostile targets. And it is a fundamental problem in solving emitter identification and threat evaluation [6, 7].

It is the informationalized and networked modern war that turns the platform centric warfare (PCW) to network centric warfare (NCW). NCW is defined as an information superiority-enabled concept of operations by Cebrowski and Garstka [8] in 1998, which generates increased combat power according to the data fusion of multisensor. It takes advantage of network linking information sources, information

users, and information transformers/planners to complete the strategic purpose. As one of the fundamental problems in NCW, data fusion techniques [9] combine the related information to achieve improved performance compared to the use of a single sensor. Three kinds of data fusion architectures are used in NCW, which are centralized fusion, distributed fusion, and mixture fusion architecture [10]. In centralized fusion architecture, the radar emitter pulses intercepted by each sensor are sent to the fusion center to process. The centralized fusion architecture could achieve the higher precision, but it is susceptible to interference. In distributed fusion architecture, each sensor processes the intercepted data firstly and then sends the result to fusion center to do further process. The distributed fusion architecture has high stability, but it could not get better precision. Mixture fusion architecture is composed of centralized fusion architecture and distributed fusion architecture, and it has the advantages of both two architectures. In this paper, we investigate the radar emitter classification problem on the basis of mixture fusion architecture in the airborne multiple ES systems to achieve appropriate stability and precision.

Radar emitter classification is a typical data clustering problem which aims at discovering the inner structure of the received radar pulses [5]. The goal of data clustering, also known as cluster analysis, is to find the natural grouping(s)

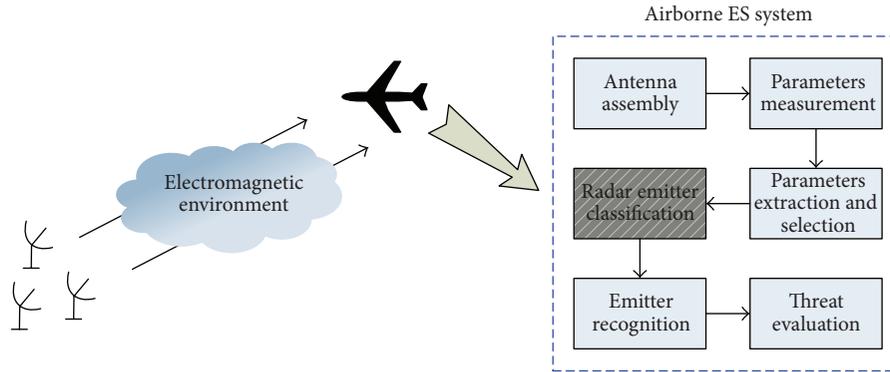


FIGURE 1: Illustration of the airborne ES system. It consists of antenna assembly, parameters measurement, radar emitter classification, parameters extraction and selection, emitter recognition, and threat evaluation.

of a set of patterns, points, or objects. It is widely used in many aspects, such as exploratory pattern-analysis, decision-making, vector quantization [11], probability density function estimation [12], and entropy maximization [13]. Clustering algorithms partition samples into a certain number of clusters (groups, subsets, or categories) by considering the internal homogeneity and external separation [14]. The patterns of data classified in the same cluster should be similar to each other, while those in different clusters should not.

Radar emitter classification has some unique challenges compared with traditional clustering problems. Firstly, the radar emitter pulses are of high dimension [5, 15]; for example, pulse sample vectors may have several hundred dimensions. Secondly, the radar emitter pulse is intercepted with uncertain probability in the electromagnetic environment [16, 17]. The number of intercepted pulses may be small, usually 10–20 percent of each radar emitter signals. Thirdly, with the development of radar technology, more high-tech and complicated radars are used in civilian application and military application [18–20]. As a result, the intercepted pulses from different radar emitters are very similar, or the pulses from the same radar emitters are totally dissimilar. Therefore, more intercepted pulses which belong to the same radar emitter are needed to obtain exact clustering result.

In previous papers, Rissanen proposes a clustering algorithm based on minimum description length (MDL) criterion in [21]. Liu et al. uses this method in emitter number detection and pulse classification [22], but this algorithm is offline. In recent years, many scholars focus their study on online clustering [23], which is used in many aspects, including wireless sensor networks [24], traffic classification [25], speaker recognition [26], fault detection [27], and data streams [28]. In [5], an online radar emitter classification algorithm is developed by Liu et al.; this algorithm uses cluster splitting and merging to modify the number of clusters when new pulse arrives. However, the drawback of this algorithm is that the algorithm needs a postprocessing procedure to improve the clustering performance after all data are available, and this algorithm is suitable for single platform to cluster the radar emitter classification. With the development of network in military application, not only the

classification result should be obtained in real time, but also the fusion of the multiple sensors is needed.

In this paper, we investigate the radar emitter classification problem and develop an online multisensor data fusion framework. This fusion framework is based on mixture fusion architecture, which makes full use of the sensors in the network to guarantee the efficiency and precision of classification result. Compared with traditional radar emitter classification algorithms, the clustering algorithm used in the proposed framework does not need prior knowledge and training process for the reason that it is based on MDL criterion to determine the number of radar emitters.

The rest of the paper is organized as follows. The background is introduced in Section 2. Section 3 gives a description of online multisensor data fusion framework, and the complexity of the proposed framework is estimated. And then the experimental results of the proposed framework in comparison with other approaches are shown in Section 4. At last, the conclusions are summarized in Section 5.

2. Background

2.1. Airborne Electronic Support System. Electronic warfare (EW) is an important part of NCW, which can be broadly defined as any military action with the objective of controlling the electromagnetic spectrum [29]. A pivotal aspect of EW is the airborne ES system, whose goal is to protect itself or the important equipment from radar-equipped threat [4]. The ES system intercepts and analyzes the radar emitter pulses, and then it sends the processed result to the control center to help the command staff to make decision. The airborne ES system usually consists of six components, as illustrated in Figure 1. A brief description of airborne ES system is given here. Firstly, the radar emitter pulses are intercepted by airborne antenna assembly with very short acquisition time and long range detection. Secondly, in parameters measurement procedure, the received pulses are digitalized and the parameters of them are measured. Thirdly, the parameters extraction and selection procedure outputs a vector which could represent the feature of the radar emitter pulse. Fourthly, the output is used to classify the radar emitter pulses in radar emitter

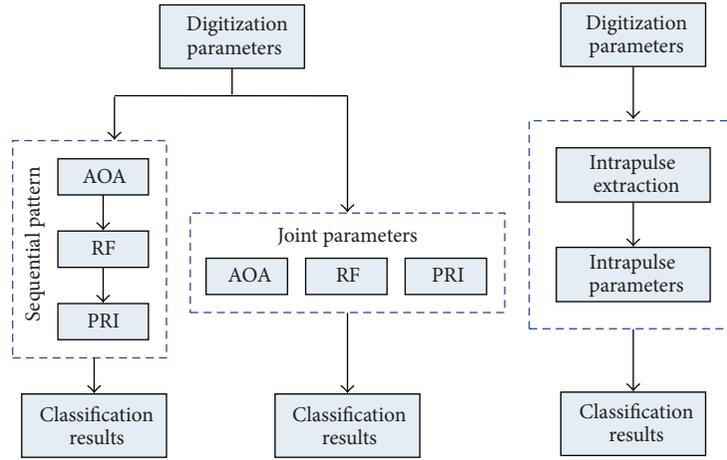


FIGURE 2: Two handling methods used in radar emitter classification.

classification. Fifthly, the classified pulses are compared with the library of known emitter characteristics to determine the possible identities of radar emitters. At last, the threat evaluation procedure evaluates threatening of the radar emitter and interprets its threat level according to its emitter type and operating mode. The airborne ES system outputs the results on an instrumentation panel in time to help the pilot make decision.

Radar emitter classification is the basic procedure to solve the emitter recognition and threat evaluation problem in airborne ES system; it closely related to the intercepted radar emitter pulses and the clustering algorithm that the airborne ES system adopted. In this paper, we focus our research on radar emitter classification of the airborne ES system after the radar emitter pulses are corrupted.

2.2. The Description of Radar Emitter Pulse. Radar emitter classification relies on the feature of the pulses radar emitter radiates. Usually the feature of radar emitter pulse can be divided into two major parts: regular pulse feature and intrinsic pulse feature. The regular feature of radar emitter pulse includes angle of arrival (AOA), time of arrival (TOA), radio frequency (RF), pulse width (PW), and pulse amplitude (PA). These five pulse features are usually called pulse description words (PDW). PDW are widely used in traditional ES system, and they are efficient in processing conventional radar emitter pulse. In recent years, the intrinsic pulse feature of radar emitter has drawn increasing attention of many scholars in the fields of electronic warfare, for the reason that the intrinsic pulse feature can reflect the characteristics of the radar emitters, which cannot be affected by other factors easily. The intrinsic pulse features are extracted from the intrapulse, such as the intrinsic pulse shape, the pulse modulation, and the higher-order spectrum component.

There are two handling methods used in radar emitter classification based on the feature of the radar emitter pulse, and they are shown in Figure 2. The left one uses regular pulse feature to classify the intercepted pulses, which is a simple and effective way. It often uses two modes: sequential

pattern and joint parameters. The AOA, RF, and PRI are usually chosen to complete the classification because the influence of electromagnetic environment on these features is small. However, with the development of the electronic technology, more high-tech and complicated radars are used in civilian and military application, which result in the fact that the method based on regular pulse features cannot classify the radar emitter pulses exactly. Therefore, another method that is based on intrinsic pulse feature is widely used in recent years, especially in the complex electromagnetic environment, because intrinsic pulse feature could reveal the intrinsic property of the radar emitter pulse. In this paper, the intrinsic pulse shape is chosen to classify the incoming emitter pulses for its easy implementation and high efficiency.

2.3. MDL Criterion. MDL criterion is used to confirm the number of radar emitters in the proposed framework. It states that [5] the best model is the one that yields the minimum code length for the dataset. Assume $Y = \{y_1, y_2, \dots, y_N\}$ be the set of N intercepted data vectors and $\hat{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$ partitions Y into \bar{K} clusters. The dataset Y can be modeled by a conditional density function $L(Y | \hat{\theta}, \hat{\alpha})$, and the description length of Y is defined as

$$L(Y, \bar{K}) = L(Y | \hat{\theta}, \hat{\alpha}) + L(\hat{\theta}, \hat{\alpha}), \quad (1)$$

where $\hat{\theta}$ is the maximum likelihood estimate of the parameter θ . $L(Y | \hat{\theta}, \hat{\alpha})$ is the negative log-likelihood function which could measure the goodness of fit between the data and the model. $L(\hat{\theta}, \hat{\alpha})$ is the penalty function which could measure the model complexity. $L(Y | \hat{\theta}, \hat{\alpha})$ and $L(\hat{\theta}, \hat{\alpha})$ are computed as

$$\begin{aligned} L(Y | \hat{\theta}, \hat{\alpha}) &= -\log f(Y | \hat{\theta}, \hat{\alpha}) \\ &= \frac{1}{2} \sum_{k=1}^{\bar{K}} N_k \log |\text{diag}(\hat{\Sigma}_k)| + \frac{MN}{2} \end{aligned}$$

$$\begin{aligned}
& + \frac{MN}{2} \log(2\pi), \\
L(\hat{\theta}, \hat{\alpha}) &= \sum_{k=1}^{\hat{K}} \log \frac{|\text{diag}(\hat{\Sigma}_0)|}{|\text{diag}(\hat{\Sigma}_k)|} + 2M \sum_{k=1}^{\hat{K}} \log N_k \\
& + \hat{K}M \log \frac{\sqrt{2}e}{6N} + \log S(N, \hat{\alpha}), \tag{2}
\end{aligned}$$

where $\text{diag}(\cdot)$ is the diagonal matrix, $|\cdot|$ is the determinant of a matrix, and $\hat{\Sigma}_0$ is the sample covariance matrix when the whole dataset is viewed as one cluster. The number of different partitions $S(N, \hat{\alpha})$ is computed by

$$\begin{aligned}
S(N, \hat{\alpha}) &= \frac{N!}{N_1! \cdots N_k! \cdots N_{\hat{K}}! (\hat{K} - 1)!} \\
& \cdot \frac{1}{m_1! \cdots m_n! \cdots N_N!}, \tag{3}
\end{aligned}$$

where N_k is the number of data vectors assigned to the k th cluster and m_n is the number of clusters containing n data vectors. Here, $N = N_1 + N_2 + \cdots + N_{\hat{K}}$, $k = 1, 2, \dots, \hat{K}$, and $n = 1, 2, \dots, N$.

It is generally acceptable in the radar emitter classification case that the noise accompanying a radar pulse vector is Gaussian. The pulses vectors are samples of a multivariate normal distribution with mean vector $\hat{\mu}_k$ and covariance matrix $\hat{\Sigma}_k$. The sample mean vector $\hat{\mu}_k$ and covariance matrix $\hat{\Sigma}_k$ are given by

$$\begin{aligned}
\hat{\mu}_k &= \frac{1}{N_k} \sum_{n=1}^{N_k} y_n(k), \\
\hat{\Sigma}_k &= \frac{1}{N_k} \sum_{n=1}^{N_k} (y_n(k) - \hat{\mu}_k)(y_n(k) - \hat{\mu}_k)^T. \tag{4}
\end{aligned}$$

Furthermore, the noise is assumed to be independent from sample to sample. Then, the conditional density function of dataset Y is a mixture of \hat{K} multivariate normal distributions with the parameter set $\hat{\theta} = \{\hat{\mu}_1, \dots, \hat{\mu}_{\hat{K}}, \hat{\Sigma}_1, \dots, \hat{\Sigma}_{\hat{K}}\}$. It is supposed that all Σ_k are diagonal given by

$$\Sigma_k = \begin{bmatrix} \sigma_{k_1}^2 & & 0 \\ & \cdots & \\ 0 & & \sigma_{k_1}^2 \end{bmatrix}, \quad k = 1, 2, \dots, \hat{K}; \tag{5}$$

Σ_k may not be diagonal in some other applications. Thus, the sample standard deviation vector $\hat{\sigma}_k$ is defined as

$$\hat{\sigma}_k^2 = \frac{1}{N_k} \sum_{n=1}^{N_k} (y_n(k) - \hat{\mu}_k)^2, \quad k = 1, 2, \dots, \hat{K}. \tag{6}$$

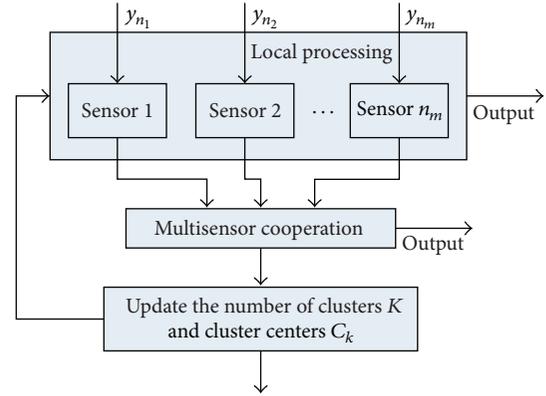


FIGURE 3: The framework of online multisensor fusion clustering. It is composed of local processing and multisensor fusion. The local processing uses simple and fast algorithms to cluster the intercepted pulses. The multisensor fusion uses the results of the local processing to cluster the radar emitter pulses further, and then the numbers of clusters and cluster centers are updated on the basis of multisensor fusion.

From the above, the MDL criterion determines the number of clusters K^* by

$$K^* = \arg \min_{1 \leq \hat{K} \leq N} L(Y, \hat{K}). \tag{7}$$

Therefore, the MDL criterion is to find the number of clusters \hat{K} which yields the minimal description length $L(Y, \hat{K})$.

3. The Proposed Framework

In this section, the online multisensor data fusion framework based on mixture fusion architecture is introduced. This framework has two procedures: local processing and multisensor fusion. The radar emitter pulse is intercepted with uncertain probability in the electromagnetic environment. Using several local sensors in the network to receive and process the radar emitter pulses from different orientations and altitudes can increase the intercept probability of radar emitter pulses. So the data fusion of multiple sensors is essential in order to make full use of the information intercepted by different sensors and improve the performance of single sensor. There are three fusion types which can be used [10]: (1) data level fusion, (2) feature level fusion, and (3) decision level fusion. In this paper, the feature level fusion type is chosen to fuse the data which is preprocessed by local sensor, because this fusion type has the advantage of high effectiveness and stability. And then we update the number of clusters and cluster centers in local sensors according to the multisensor fusion result.

Besides, both the results of local processing and multisensor fusion can be output in time. The local processing can be used to send threatening radar warning information to the control center because of its real-time property, and the multisensor fusion can be used for deep analysis of radar emitter pulses due to its accuracy. The framework of the proposed method is shown in Figure 3.

3.1. Local Processing. In each local sensor, the preprocessing should be done before the multisensor fusion. The preprocessing concludes the feature extraction and local sensor clustering.

3.1.1. Signal Model and Features Extraction. Suppose $x_n(t, a_n)$ is the n th received pulse; it can be presented as follows [22]:

$$x_n(t, a_n) = \eta_n a_{\alpha_n}(t - \tau_n) e^{j[\phi_n + \omega_n(t - \tau_n) + \phi_{\alpha_n}(t - \tau_n)]} + v_n(t - \tau_n), \quad n = 1, 2, \dots, N, \quad (8)$$

where α_n is the association parameter; for example, $\alpha_n = k$ indicates that the n th pulse radiates from the k th emitter. η_n and ϕ_n present the initial amplitude and phase of the received pulse, respectively. ω_n is the carrier frequency, and τ_n is the time delay of the received pulse. $a_{\alpha_n}(t)$ and ϕ_{α_n} present the original envelope and phase for the n th received pulse, respectively. $v_n(t)$ is the Gaussian noise accompanying the n th received pulse.

In the proposed algorithm, not all these parameters carry useful information in confirming the number of radar emitters and clustering the intercepted pulses, such as η_n , ϕ_n , τ_n , and ω_n . So, the unnecessary parameters are removed in preprocessing and the signal model can be expressed [5] as follows:

$$y_n(t, a_n) = a_{\alpha_n}(t) e^{j\phi_{\alpha_n}(t)} + \tilde{v}_n(t), \quad n = 1, 2, \dots, N. \quad (9)$$

Since preprocessing should be done in discrete time in practical, (9) is rewritten as follows:

$$y_n(mT, a_n) = s_{\alpha_n}(mT) + \tilde{v}_n(mT), \quad (10)$$

where T is the sampling interval of the pulses. As a result, the data vectors can be expressed in vector form:

$$y_n(a_n) = s_{\alpha_n} + \tilde{v}_n, \quad (11)$$

where $y_n(a_n) = [y_n(T, a_n), \dots, y_n(M'T, a_n)]^T$, $s_{\alpha_n} = [s_{\alpha_n}(T), \dots, s_{\alpha_n}(M'T)]^T$, and $\tilde{v}_n = [\tilde{v}_n(T), \dots, \tilde{v}_n(M'T)]^T$. M' is the number of sample points for each pulse. At last, each data vector is formed by its real part and its imaginary part before the data clustering, which is presented by

$$\text{Data}_{\text{vector}} = [\text{real}(y_n^T), \text{imag}(y_n^T)]^T. \quad (12)$$

Thus, $\text{Data}_{\text{vector}}$ is the feature expression of the radar emitter pulses used in this paper.

3.1.2. Local Sensor Clustering. The main task of the local processing is local sensor clustering; it groups the intercepted pulse y_{n_i} to the nearest cluster center quickly. The procedure of local processing is shown in Figure 4. y_{n_i} is a radar pulse radiated from the i th emitter interrupted by the local sensor, and it is represented in the form of $\text{Data}_{\text{vector}}$. After implementing this procedure, the results of local processing could be used to send threatening radar warning information to the local control center in real time because of the simple and fast property of the local processing.

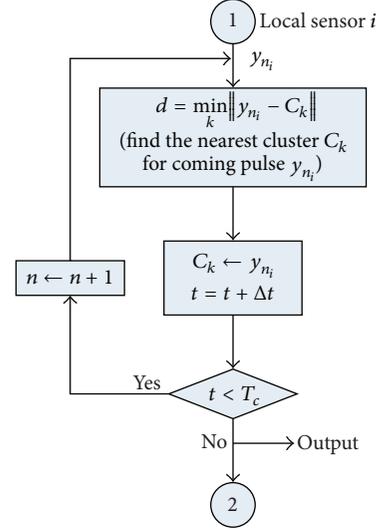


FIGURE 4: The procedure of local processing. In each sensor, the coming pulse y_{n_i} is attributed to the nearest cluster center C_k according to the minimum distance. If the time increment counter reaches a preset threshold T , the results are sent to the fusion center. In addition, the results would be output to the local control center in time if necessary.

The key problem in classification is to find the similarity between two radar emitter pulses. The distance criterion is an efficient way to measure the difference between different pulses [30]. In feature space, the distance between two patterns is usually small when they belong to the same class; on the contrary, the distance is usually big when they belong to different classes. Particularly, the Euclidean distance [31, 32] is adopted in this paper, which is described as follows:

$$d(y, C_k) = \|y - C_k\| = \left(\sum_{m=1}^M (y_m - C_{km})^2 \right)^{1/2}, \quad (13)$$

$$y = \{y_1, \dots, y_M\}, \quad C_k = \{C_{k1}, \dots, C_{kM}\},$$

where C_k is the cluster center and pulse y is assigned to the cluster that has the minimum distance value. We calculate the initial cluster center by k -means for the radar emitter pulses which are first intercepted by the local sensor, and then each cluster center is updated by multisensor fusion result.

3.2. Multisensor Fusion

3.2.1. Data Fusion Model. If the time increment counter reaches a preset threshold T , the multisensor data fusion starts to work according to the results of local processing, the numbers of clusters and cluster centers are updated according to the multisensor fusion result. Two fusion models are proposed as follows.

The first one uses the same kind of platforms to do data fusion, which is shown in Figure 5 (we choose airborne ES system to illustrate in this paper). The fusion unit is composed of several fighters called local sensor which fly in formation, and each local sensor processes the intercepted pulses and

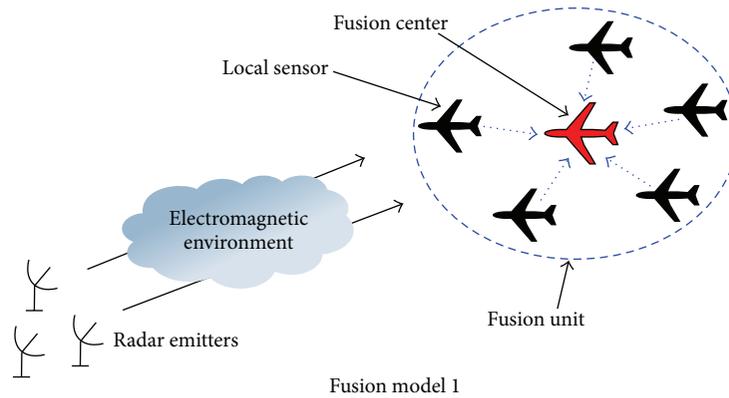


FIGURE 5: The same kind of platforms fusion model. In fusion model 1, the radar emitter pulses are intercepted by different local sensors from different orientations and altitudes in the fusion unit, and then the processed pulses are sent to the fusion center to do data fusion.

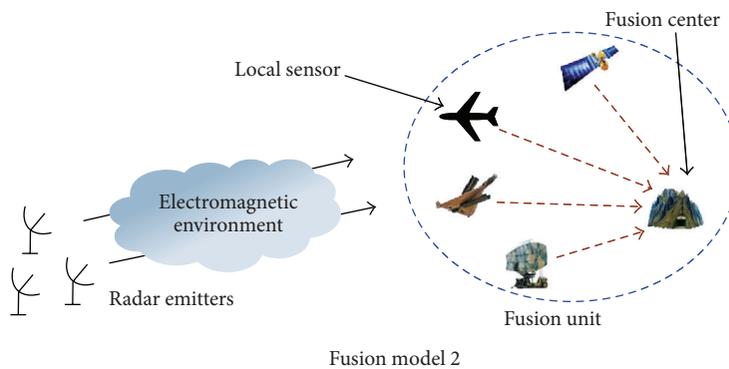


FIGURE 6: Different kinds of platforms fusion model. The fusion unit in fusion model 2 could be composed by several kinds of sensors, such as fighter, satellite, unmanned aerial vehicle (UAV), or passive radar. The radar emitter signal intercepted by several kinds of sensors from different space and the initial classification results of these signals are sent to the fusion center.

sends the results to the fusion center. This unit is the smallest unit or the basic unit, which could be used in other data fusion models. In reality, there are often large amount of sensors, and it is usually a huge computation burden to fuse all of them together, so we could use some specific criteria to divide the sensors into several basic units and then do the multisensor fusion with the proposed framework.

Another model uses several different kinds of platforms from different space to do data fusion, which is shown in Figure 6. Moreover, it can also consist of several basic units similar to fusion model 1. This could increase the intercept probability of radar emitter pulses and guarantee the accuracy of the classification.

3.2.2. Clustering. It is common that different classification results may be obtained from different sensors even though the pulses come from the same radar emitters, so a multisensor data fusion step is necessary in the radar emitter classification. In the proposed framework, a fast but coarse result is obtained in the local sensors, while the multisensor fusion scheme can update the number of clusters and help improve the classification accuracy and data integrity. It contains three steps as shown in Figure 7, and the details can be described as follows.

Step 1. Combine the same clusters which come from local processing and update the cluster centers using k -means.

Step 2. Detect whether each cluster C_i should be split or not by the function detector. If it should not be split, cluster C_i keeps constant; otherwise the cluster will be split into two new clusters and go to Step 3.

Step 3. Use detector to detect whether the two new groups should be regrouped or merged with other existing clusters. Four cases are considered here, which are shown in Figure 8. The first case is that both the new clusters find the nearest cluster to them and regroup as two new clusters. The second and the third cases are similar; that is, only one of new clusters splits into two new groups after combining with its nearest cluster, while another cluster finds its nearest cluster and they merge as one cluster. The last case is that both the two new clusters find their nearest clusters and merge as one cluster.

The termination condition of the multisensor data fusion is that all incoming clusters have been processed. The flowchart of Steps 2 and 3 is shown in Figure 9.

As shown in Figure 9, we need to detect if one cluster should be split into two new clusters and if two clusters

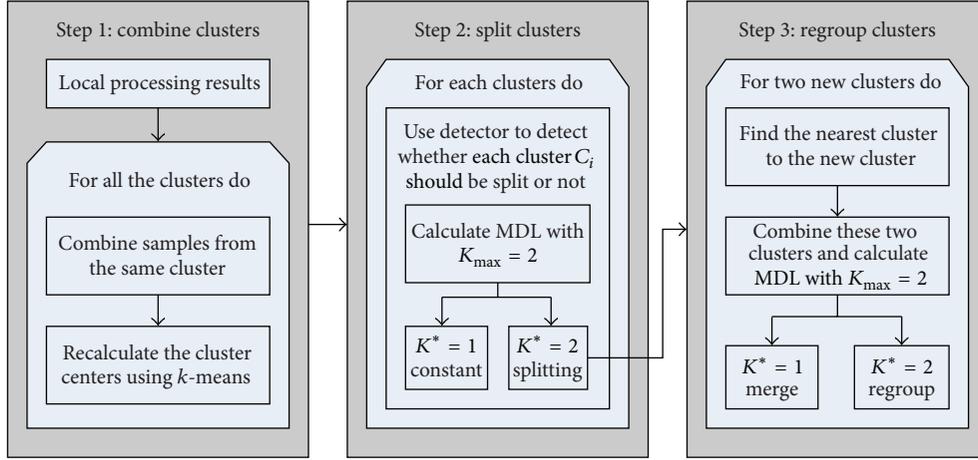


FIGURE 7: The main procedures of clustering.

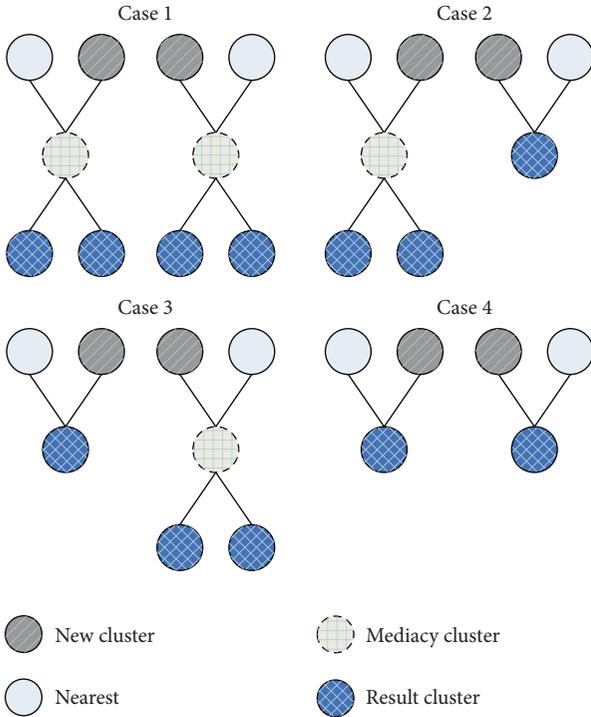


FIGURE 8: The procedure of regroup or merge.

should be merged into one cluster or regrouped into two new clusters. These procedures are based on the detection function which is called detector with the following form:

$$[K^*, C_{x1}, C_{x2}] = \text{detector}(x), \quad (14)$$

where x is input and K^* , C_{x1} , and C_{x2} are the outputs. K^* is the number of clusters according to MDL criterion with $K_{\max} = 2$. C_{x1} and C_{x2} are two new clusters produced by detector.

There are two cases considered here; the first case is that input x comes from one existing cluster (marked with * in Figure 9); then if output $K^* = 1$, x will not be split; otherwise

it will be split into two clusters C_{x1} and C_{x2} . Another case is that input x comes from two existing clusters (marked with Δ in Figure 9); if $K^* = 1$, x will be merged as one cluster; otherwise it will be regrouped into two new clusters C_{x1} and C_{x2} .

In (9), no matter the input data x comes from one cluster or two clusters, the cluster center $\hat{\mu}$ and standard deviation $\hat{\sigma}$ should be computed as one cluster in function detector. Regarding the case when x comes from two clusters, the two new cluster centers are initialized as

$$\begin{aligned} \hat{\mu}_1 &\leftarrow \hat{\mu} - \hat{\sigma}, \\ \hat{\mu}_2 &\leftarrow \hat{\mu} + \hat{\sigma}. \end{aligned} \quad (15)$$

And this process will not be terminated until it is convergent (when $\hat{\mu}_1$ and $\hat{\mu}_2$ keep stable).

3.3. Computational Complexity. In this section, the computational complexity of the proposed algorithm is estimated. Assume N_i radar pulses with M dimension are intercepted and classified into K clusters by i th sensor in T time. Then the complexity of the local processing procedure is $O(MKN_i)$ for i th sensor.

Considering the multisensor data fusion procedure, the minimum fusion unit consists of m sensors and the total number of radar pulses is N ; the number of iterations for the samples structure to converge is N_t . There are three steps which need to be estimated in this procedure. First, the computational complexity of the cluster centers update step after combining the data of the minimum fusion unit is $O(MNK_n)$, where n is the number of iterations needed until the convergence of this step. Second, the complexity of the split/regroup/merge step is $O(21MNN_t)$ on the worst case (all the clusters need to split and execute the detector operation 3 times for each cluster). At last, the new cluster centers need to be calculated and updated for each sensor, and the computational complexity of this step is $O(MN)$. Therefore, take the worst case of the local processing which is $O(MKN)$ into account; the computational complexity of the

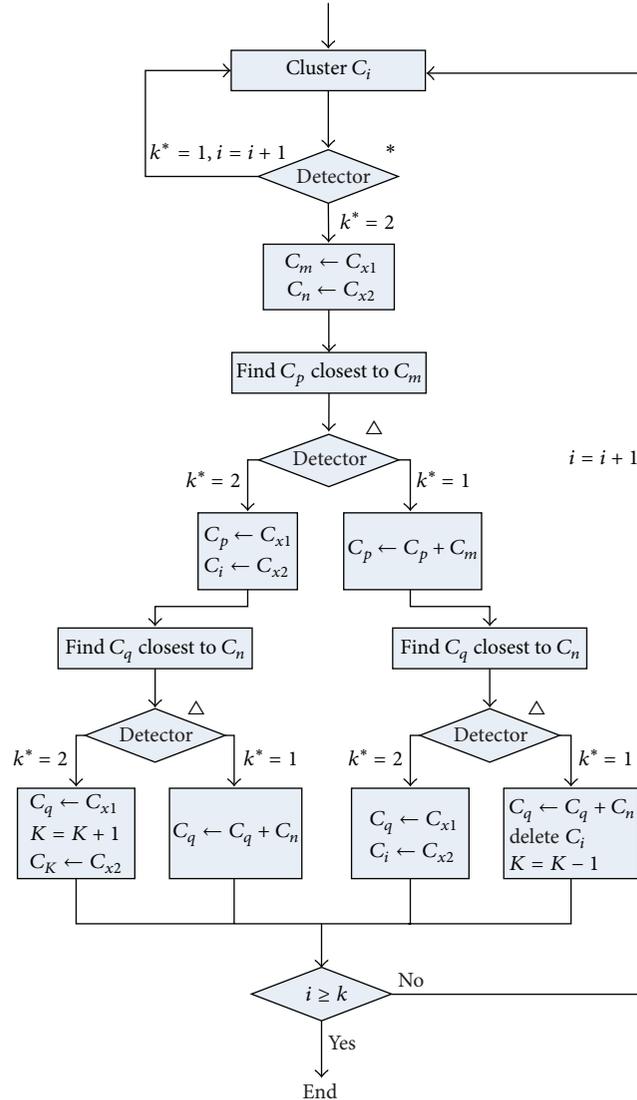


FIGURE 9: The flowchart of clustering.

proposed algorithm is $O((MNK(n+1)+MN+21MNN_t)t/T)$ in period t .

4. Experiments

In this section, the proposed method is applied to radar emitter classification. We use Gaussian datasets and intrapulse datasets to test the proposed algorithm. In order to simulate the experiment realistically, the radar scanning modes are simulated in signal transmission, such as mechanical scanning and phased array scanning [18], to represent the stochastic characteristic of the electromagnetic environment. In addition, the passive sensors do not know the number of radar emitters and which radar the intercepted pulses belong to.

4.1. Experiment Setting. In this paper, we compare the proposed algorithm with the algorithm mentioned in [5] (denoted as online MDL in the following part). Since the proposed algorithm is a multisensor data fusion framework

and online MDL is a singular sensor model, the online MDL is implemented in each sensor and compared with the proposed algorithm in order to see the importance of data fusion.

The algorithms used in this paper are all written in MATLAB; the experiments have been performed on an Intel® Core™ i7-4770 CPU @3.40 GHz, 3.46 GB memory. The operating system is Windows 7 PC and the compiler is MATLAB R2011a.

4.2. Experiments on Gaussian Datasets. Firstly, the proposed framework is applied to Gaussian datasets [5]. The Gaussian datasets are with the number of clusters ranging from 10 to 25 with interval 5 and each cluster has 2000 samples. Both the centers of the datasets and the samples in each cluster follow Gaussian distribution. The number of sensors in the network is set to 5, 8, and 10. All the algorithms run 20 independent times on each test problem.

The simulation results on datasets with different number of clusters and sensors are shown in Figure 10. The four

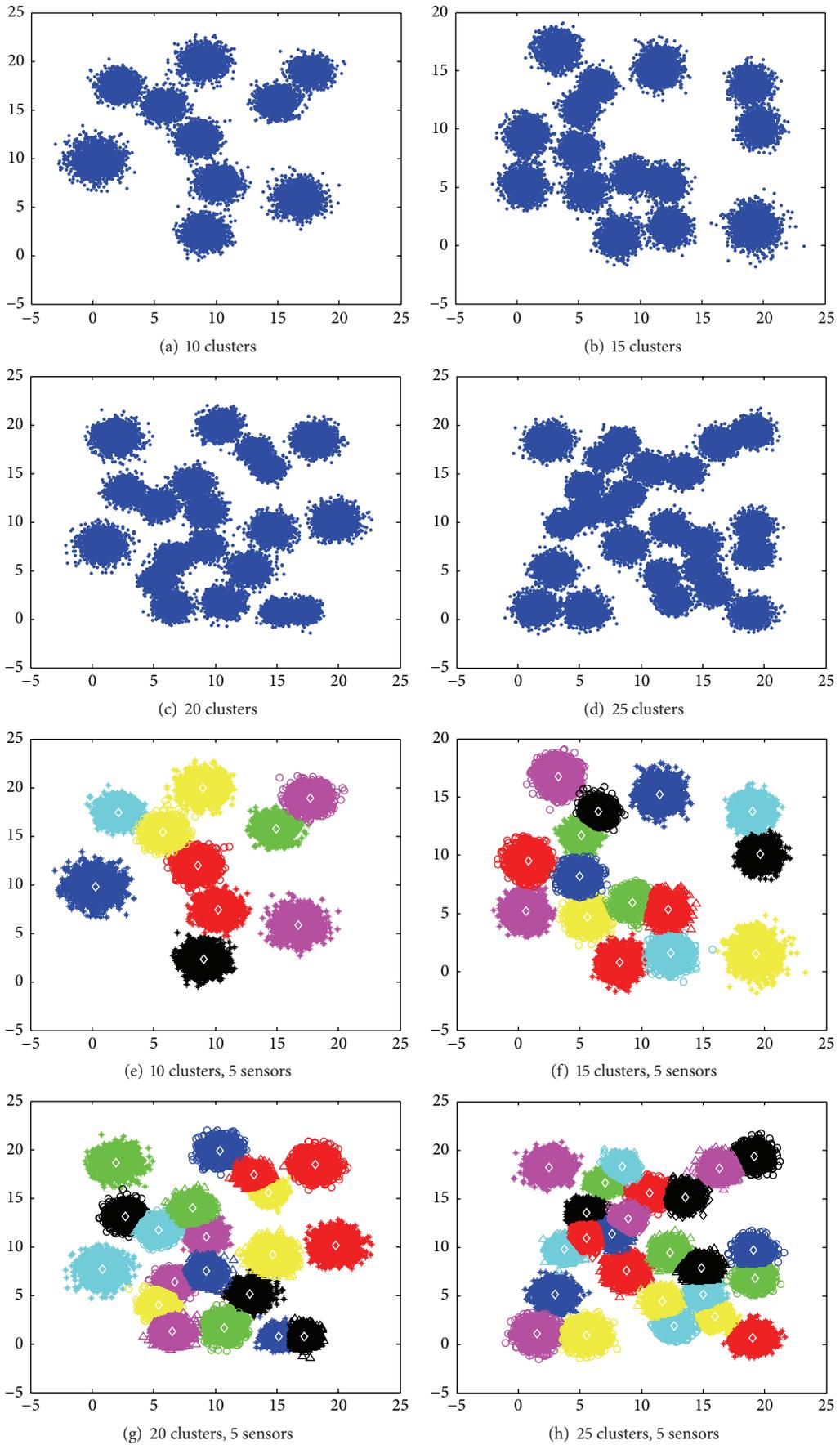


FIGURE 10: Continued.

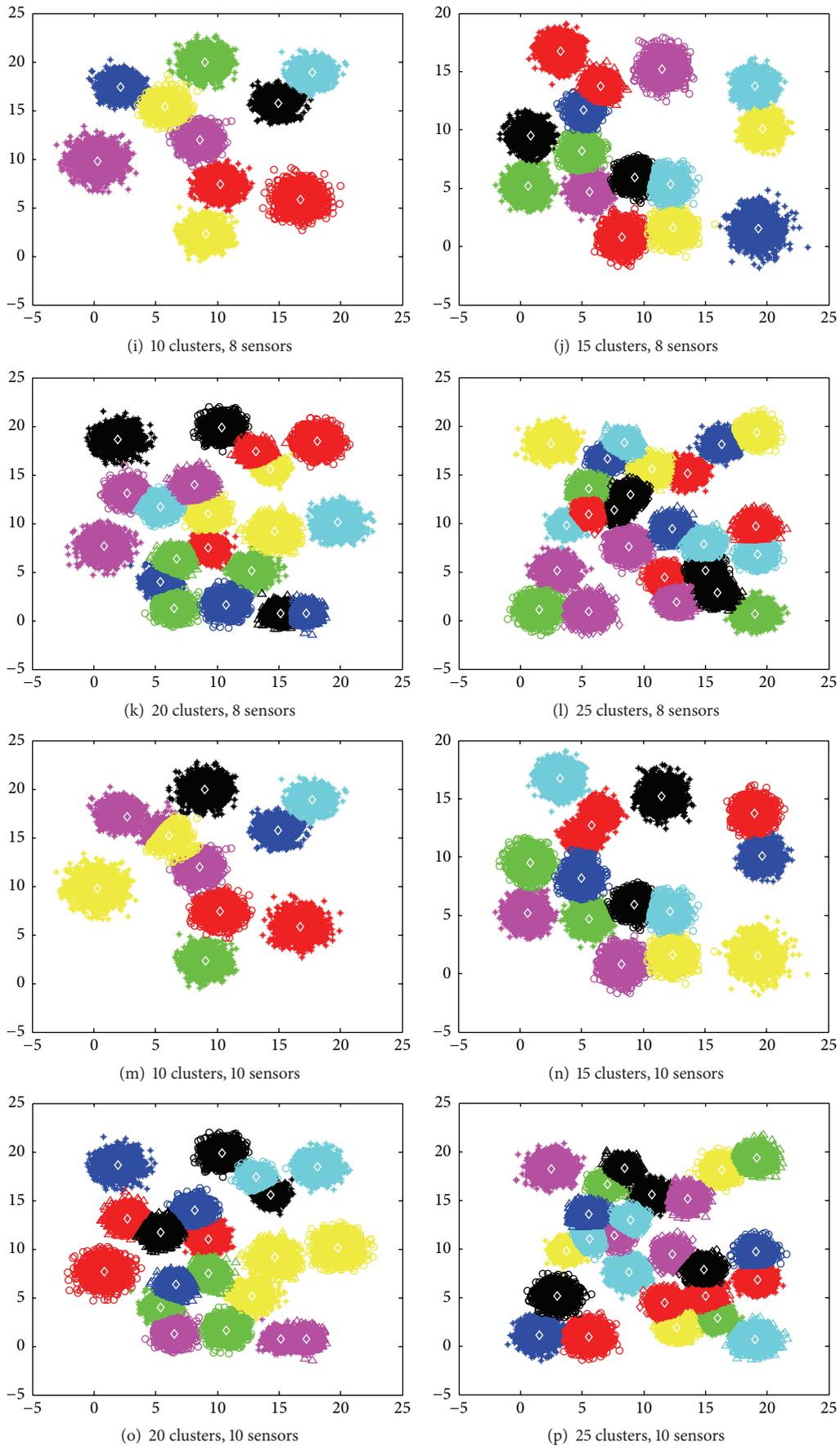


FIGURE 10: Classification results of the proposed framework in different number of clusters and sensors.

TABLE 1: Classification accuracy of 5 sensors.

C_N	Online MDL					Prop.
	S_1	S_2	S_3	S_4	S_5	
10	86.29%	79.65%	87.85%	90.29%	93.97%	99.54%
15	73.78%	80.42%	85.08%	84.08%	89.78%	99.42%
20	87.37%	73.95%	93.55%	83.83%	84.43%	99.01%
25	85.71%	71.04%	91.60%	80.51%	88.93%	98.49%

TABLE 2: Classification accuracy of 8 sensors.

C_N	Online MDL								Prop.
	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	
10	84.80%	80.91%	86.53%	74.43%	80.43%	94.63%	92.43%	85.77%	95.05%
15	78.30%	80.69%	73.14%	86.06%	79.39%	74.14%	77.80%	76.57%	97.60%
20	73.87%	83.52%	79.28%	88.17%	82.06%	86.53%	70.08%	71.59%	99.01%
25	88.60%	68.42%	95.19%	92.99%	86.58%	73.80%	88.09%	76.48%	98.47%

TABLE 3: Classification accuracy of 10 sensors.

C_N	Online MDL										Prop.
	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	
10	88.06%	82.69%	81.15%	94.87%	84.14%	86.36%	83.13%	87.11%	92.80%	88.57%	94.54%
15	74.10%	80.05%	84.82%	84.79%	88.86%	76.19%	89.02%	85.23%	84.50%	86.87%	97.23%
20	85.72%	62.80%	71.58%	72.57%	74.62%	74.64%	77.82%	82.21%	81.68%	79.08%	98.51%
25	78.40%	83.21%	90.15%	86.62%	65.75%	73.03%	88.22%	81.50%	81.27%	82.80%	98.48%

original datasets are tabulated in Figures 10(a)–10(d) and the results of the proposed framework with different number of sensors or clusters are shown in Figures 10(e)–10(p).

As shown in Figure 10, compared with original datasets, the proposed algorithm shows its excellent clustering ability in dealing with different number of clusters and sensors except two cases. One case is Figure 10(m), which has 10 clusters and 10 sensors; some samples of one cluster are grouped into another adjacent cluster. The other case is Figure 10(n), which has 15 clusters and 10 sensors; the two clusters are regarded as one cluster.

In addition, the average classification accuracy is shown in Tables 1–3 for further analysis. *Classification accuracy* is defined as the percentage of members in a true cluster classified into one cluster. In these tables, C_N and Prop. present the number of clusters and the proposed method, respectively, S_i represents the i th sensor, and the average *classification accuracies* are tabulated to compare.

As can be seen from these tables, three observations can be obtained: (1) the proposed multisensor data fusion framework has a better performance than singular sensor classification framework, since not only the proposed algorithm uses several sensors from different orientations and altitudes to receive the radar emitter pulses which could guarantee the data integrity but also multisensor data fusion increases the classification accuracy; (2) for singular sensor classification algorithm, we cannot see the relationship between the classification accuracy and the number of clusters or sensors, due to the uncertainty of the interception of radar emitter pulses

TABLE 4: Time cost of 5 sensors.

C_N	Online MDL					Total	Prop.
	S_1	S_2	S_3	S_4	S_5		
10	0.36 s	0.31 s	0.32 s	0.33 s	0.40 s	1.72 s	1.88 s
15	0.58 s	0.55 s	0.63 s	0.61 s	0.54 s	2.91 s	3.20 s
20	0.83 s	0.82 s	0.85 s	0.84 s	0.78 s	4.24 s	4.48 s
25	1.12 s	1.08 s	1.17 s	1.10 s	1.05 s	5.52 s	6.32 s

in electromagnetic environment; (3) the proposed algorithm has a robust performance no matter how the number of clusters or sensors changes.

In order to discuss the stability of the proposed algorithm, the boxplots of the classification accuracy on each dataset are shown in Figure 11. The last column of each boxplot is the result of the proposed algorithm and the others are results of online MDL algorithm. It can be easily seen that online MDL algorithm fluctuates strongly in each simulation, but the performance of the proposed algorithm is stable except in Figures 11(b) and 11(c). From this figure, the stability of proposed algorithm is demonstrated.

Furthermore, we test the time cost of the proposed algorithm compared with online MDL algorithm, and the result is shown in Tables 4–6.

By analyzing these tables in detail, we can see the following (1) when the number of clusters keeps constant, the time cost of each sensor decreases with the increasing of the number of sensors, because each sensor handles less

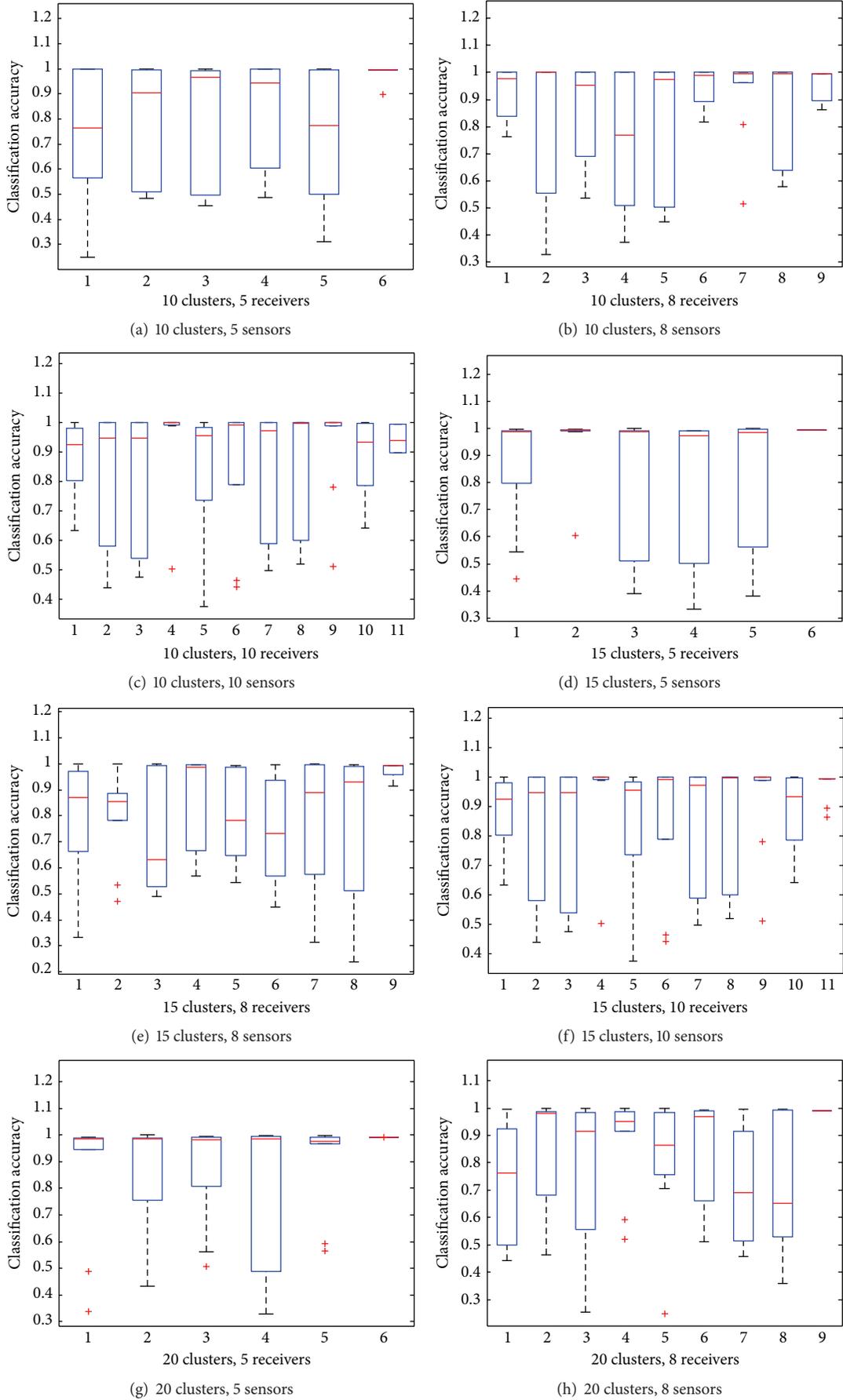


FIGURE II: Continued.

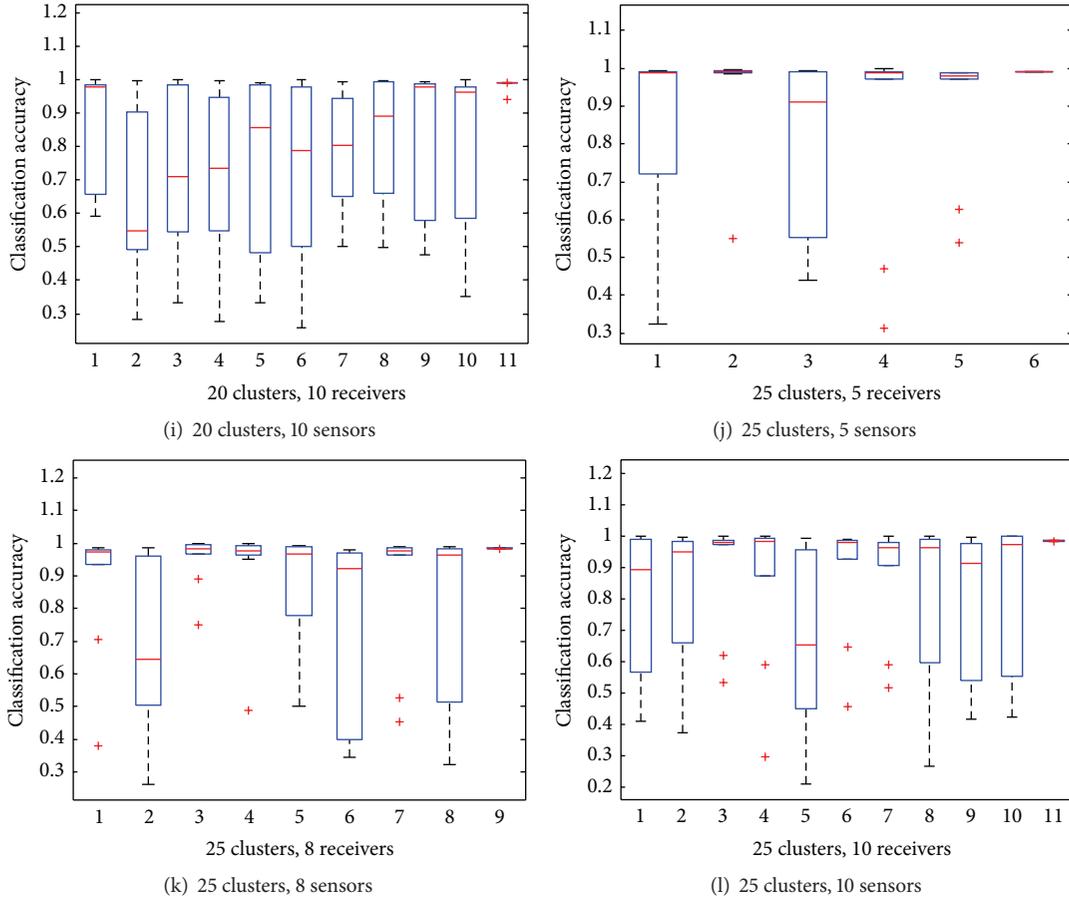


FIGURE 11: Boxplot of classification accuracy.

TABLE 5: Time cost of 8 sensors.

C_N	Online MDL									Prop.
	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	Total	
10	0.18 s	0.13 s	0.15 s	0.16 s	0.15 s	0.16 s	0.14 s	0.16 s	1.23 s	1.75 s
15	0.28 s	0.26 s	0.29 s	0.32 s	0.27 s	0.25 s	0.24 s	0.30 s	2.21 s	3.06 s
20	0.43 s	0.40 s	0.39 s	0.47 s	0.48 s	0.43 s	0.52 s	0.41 s	3.53 s	4.10 s
25	0.61 s	0.63 s	0.58 s	0.60 s	0.59 s	0.67 s	0.60 s	0.55 s	4.74 s	5.27 s

TABLE 6: Time cost of 10 sensors.

C_N	Online MDL										Prop.	
	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}		Total
10	0.13 s	0.12 s	0.09 s	0.15 s	0.13 s	0.12 s	0.11 s	0.13 s	0.12 s	0.10 s	1.20 s	1.55 s
15	0.20 s	0.21 s	0.26 s	0.18 s	0.23 s	0.21 s	0.20 s	0.19 s	0.20 s	0.24 s	2.12 s	2.66 s
20	0.40 s	0.31 s	0.32 s	0.38 s	0.42 s	0.37 s	0.36 s	0.41 s	0.35 s	0.35 s	3.67 s	4.22 s
25	0.45 s	0.36 s	0.38 s	0.40 s	0.48 s	0.39 s	0.41 s	0.43 s	0.44 s	0.41 s	4.15 s	5.66 s

samples; (2) when the number of sensors keeps constant, the time cost of each sensor increases with the increasing of intercepted sample number. But no matter in which case, when we sum up all the time costs of each sensor, the total time of all sensors is almost equal to the proposed algorithm when handling the same Gaussian dataset, which can be explained by the time complexity section in this paper.

However, the proposed algorithm gets better performance than online MDL algorithm.

4.3. *Experiments on Intrapulse Datasets.* In this part, we apply the proposed algorithm to intrapulse datasets [5, 22]. Figure 12 shows the amplitude and phase profiles of 120 compressed preprocessed radar pulse vectors. These pulses

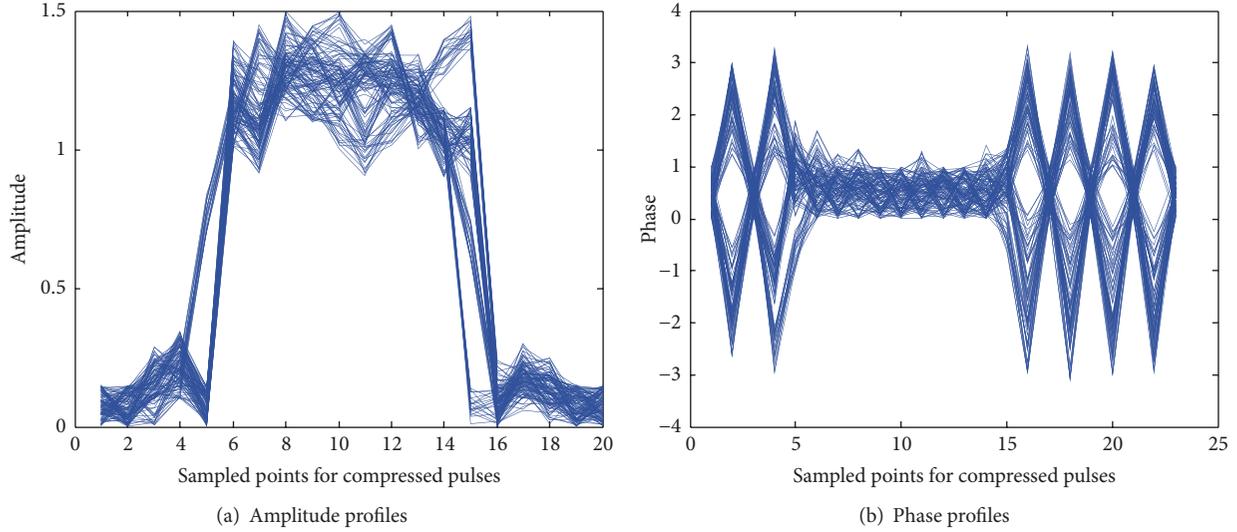


FIGURE 12: Amplitude and phase profiles of 120 compressed preprocessed radar pulses.

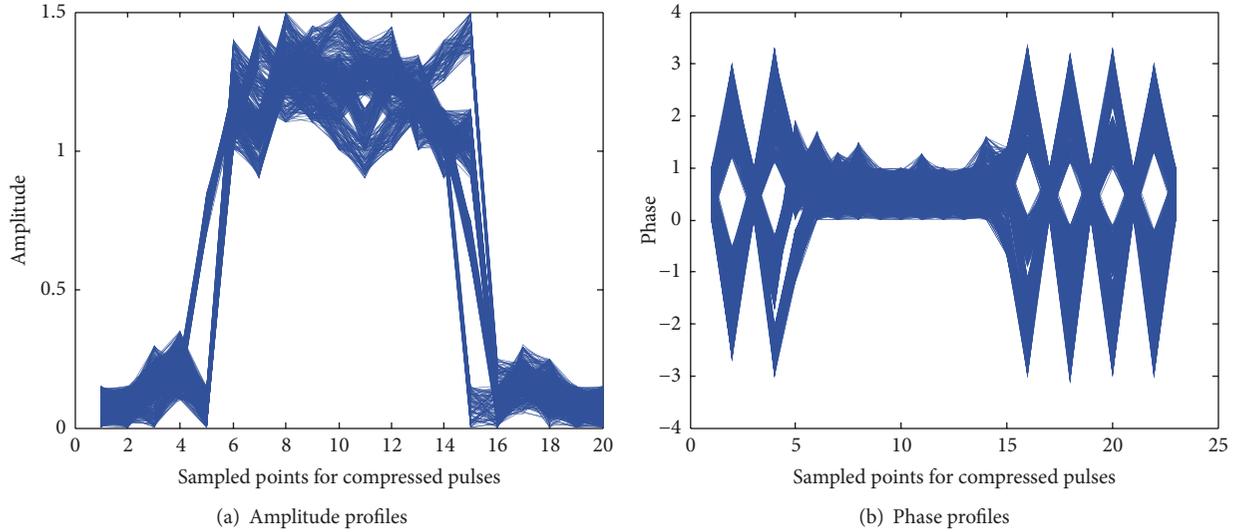


FIGURE 13: Amplitude and phase profiles of 600 compressed preprocessed radar pulses.

TABLE 7: Performance results for 120 pulses.

Algorithm	Parameters	Classification accuracy	Correct numbers in clusters	Time cost
The proposed	6 clusters	99.17%	20, 20, 19, 20, 20, 20	0.16 s
Online MDL	6 clusters	99.17%	20, 20, 19, 20, 20, 20	0.11 s

are generated from six different radar emitters; each of them transmits 20 pulses with the same simulation parameter settings in [33]. We use 5 sensors' fusion to compare with online MDL algorithm. The online MDL algorithm is used in each sensor and the best result is chosen to compare. The simulation results are shown in Table 7.

As shown in Table 7, the proposed algorithm and online MDL algorithm both achieve a classification accuracy of 99.17%. The time cost of the proposed algorithm is slightly worse than online MDL algorithm.

Figure 13 shows the amplitude and phase profiles of 600 compressed preprocessed radar pulses. These pulses are generated from six different radar emitters; 100 pulses are transmitted in each emitter. The simulation results are shown in Table 8.

As can be seen from Table 8, when the number of pulses increases, the proposed algorithm shows its superiority to online MDL algorithm. The classification accuracy of the proposed algorithm is better than online MDL algorithm. In the case of time cost, the proposed algorithm gets a neck to

TABLE 8: Performance results for 600 pulses.

Algorithm	Parameters	Classification accuracy	Correct numbers in clusters	Time cost
The proposed	6 clusters	99.50%	100, 100, 98, 100, 99, 100	0.26 s
Online MDL	6 clusters	91.30%	98, 93, 87, 96, 88, 86	0.22 s

neck performance with online MDL algorithm. Above all, a conclusion can be drawn that the proposed framework has a better performance than online MDL algorithm.

5. Conclusion

The high-tech and complicated radars used in civilian and military application increase the difficulty in radar emitter classification. The singular sensor shows its limitations because it could not get the integrity radar emitter data and the accurate classification result. As a result, an online multisensor data fusion framework, which is composed of local processing and multisensor fusion, is proposed for radar emitter classification in this paper. This framework is suitable for NCW for the reason that a fast but coarse result is obtained from local processing to send threatening radar warning information, and a more satisfying result is obtained from multisensor fusion to analyze the radar emitter pulse deeply. The number of clusters can be obtained and updated dynamically by using MDL criterion rather than set-up in advance. It also enables the framework to detect the unknown threat or new kind of radar emitter pulses in time. The experimental results show the efficiency and stability of the proposed algorithm. Moreover, how to apply it to radar emitter recognition and threat evaluation is our future work.

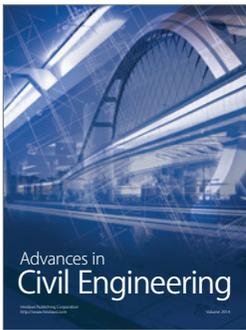
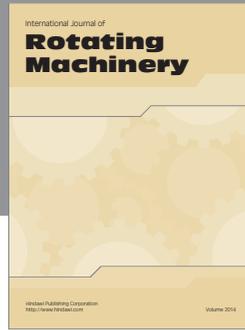
Competing Interests

The authors declare that they have no competing interests.

References

- [1] H. D. Liu, Y. Hongqi, S. Zhaolin, and D. Jietao, "Multi-function radar emitter identification based on stochastic syntax-directed translation schema," *Chinese Journal of Aeronautics*, vol. 27, no. 6, pp. 1505–1512, 2014.
- [2] G. Wang, S. Tan, C. Guan, N. Wang, and Z. Liu, "Multiple model particle filter track-before-detect for range ambiguous radar," *Chinese Journal of Aeronautics*, vol. 26, no. 6, pp. 1477–1487, 2013.
- [3] W. Zhang, Y. Xiong, P. Wang, J. Wang, and B. Tang, "An approach for parameter estimation of combined CPPM and LFM radar signal," *Chinese Journal of Aeronautics*, vol. 26, no. 4, pp. 986–992, 2013.
- [4] A. Wang and V. Krishnamurthy, "Signal interpretation of multi-function radars: modeling and statistical signal processing with stochastic context free grammar," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1106–1119, 2008.
- [5] J. Liu, J. P. Y. Lee, L. Li, Z.-Q. Luo, and K. M. Wong, "Online clustering algorithms for radar emitter classification," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 27, no. 8, pp. 1185–1196, 2005.
- [6] A. E. Spezio, "Electronic warfare systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, pp. 633–644, 2002.
- [7] D. S. Alberts, J. J. Garstka, F. P. Stein et al., "Network centric warfare: developing and leveraging information superiority," in *Command and Control Research Program (CCRP)*, pp. 78–79, US DoD, 2000.
- [8] A. K. Cebrowski and J. J. Garstka, "Network-centric warfare: its origin and future," *US Naval Institute Proceedings*, pp. 28–35, 1998.
- [9] N.-E. E. Faouzi, H. Leung, and A. Kurian, "Data fusion in intelligent transportation systems: progress and challenges—a survey," *Information Fusion*, vol. 12, no. 1, pp. 4–10, 2011.
- [10] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.
- [11] V. S. Cherkassky and F. Mulier, "Learning from data: concepts, theory, and methods," in *Reformulation: Nonsmooth Piecewise Smooth Semismooth & Smoothing Methods*, Kluwer Academic Publishers, 1998.
- [12] S. C. Kothari and H. Oh, "Neural networks for pattern recognition," *Agricultural Engineering International the Cigr Journal of Scientific Research & Development Manuscript Pm*, vol. 12, no. 5, pp. 1235–1242, 1995.
- [13] B. Fritzke, *Some Competitive Learning Methods*, Artificial Intelligence Institute, 1997.
- [14] P. Hansen and B. Jaumard, "Cluster analysis and mathematical programming," *Mathematical Programming*, vol. 79, no. 1, pp. 191–215, 1997.
- [15] G. Zhang and Z.-D. Zhu, "Application of fuzzy C-mean cluster algorithm on clutter tracking," *Chinese Journal of Aeronautics*, vol. 15, no. 1, pp. 44–48, 2002.
- [16] F. A. Butt and M. Jalil, "An overview of electronic warfare in radar systems," in *Proceedings of the International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAECE '13)*, pp. 213–217, IEEE, Konya, Turkey, May 2013.
- [17] Z. Qingzhu, Z. Jun, Z. Yanbo, and L. Wei, "Navigation accuracy category-position models and estimate position uncertainty calculations for TIS-B system," *Chinese Journal of Aeronautics*, vol. 22, no. 4, pp. 419–425, 2009.
- [18] N. Visnevski, V. Krishnamurthy, A. Wang, and S. Haykin, "Syn-tactic modeling and signal processing of multifunction radars: a stochastic context-free grammar approach," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 1000–1025, 2007.
- [19] H. Wang, L. Shi, Y. Wang, and D. Ben, "A novel target detection approach based on adaptive radar waveform design," *Chinese Journal of Aeronautics*, vol. 26, no. 1, pp. 194–200, 2013.
- [20] X. Zeng, B. Tang, and Y. Xiong, "Interception algorithm of S-cubed signal model in stealth radar equipment," *Chinese Journal of Aeronautics*, vol. 25, no. 3, pp. 416–422, 2012.
- [21] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.
- [22] J. Liu, S. Gao, Z. Q. Luo et al., "The minimum description length criterion applied to emitter number detection and pulse

- classification,” in *Proceedings of the 9th IEEE SP Workshop on Statistical Signal and Array Processing*, pp. 172–175, Portland, Ore, USA, September 1998.
- [23] W. Barbakh and C. Fyfe, “Online clustering algorithms,” *International Journal of Neural Systems*, vol. 18, no. 3, pp. 185–194, 2008.
- [24] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks,” in *Proceedings of the International Conference on Network-Based Information Systems*, pp. 2826–2841, Elsevier, Regensburg, Germany, 2007.
- [25] J. Erman, M. Arlitt, and A. Mahanti, “Traffic classification using clustering algorithms,” in *Proceedings of the SIGCOMM Workshop on Mining Network Data (MineNet '06)*, pp. 281–286, Pisa, Italy, September 2006.
- [26] D. Lilt and F. Kubala, “Online speaker clustering,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, vol. 1, IEEE, May 2004.
- [27] P. A. D. F. R. Hojen-Sorensen, N. de Freitas, and T. Fog, “On-line probabilistic classification with particle filters,” in *Proceedings of the IEEE Signal Processing Society Workshop Neural Networks for Signal Processing X*, vol. 1, pp. 386–395, IEEE, Sydney, Australia, 2000.
- [28] J. Beringer and E. Hüllermeier, “Online clustering of parallel data streams,” *Data & Knowledge Engineering*, vol. 58, no. 2, pp. 180–204, 2006.
- [29] D. C. Schleher, “Electronic warfare in the information age,” *Issues in Science & Technology*, vol. 12, no. 1, pp. 59–66, 1999.
- [30] G. A. Montazer, H. Khoshniat, and V. Fathi, “Improvement of RBF neural networks using Fuzzy-OSD algorithm in an online radar pulse classification system,” *Applied Soft Computing*, vol. 13, no. 9, pp. 3831–3838, 2013.
- [31] J. Mao and A. K. Jain, “A self-organizing network for hyperellipsoidal clustering (HEC),” *IEEE Transactions on Neural Networks*, vol. 7, no. 1, pp. 16–29, 1996.
- [32] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data clustering: a review,” *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.
- [33] P. K. M. Wong, Z. Q. Luo, and J. Liu, “Radar emitter classification using intrapulse data,” *AEÜ—International Journal of Electronics and Communications*, vol. 53, no. 6, pp. 324–332, 2000.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

