Sui, Jinping; Liu, Zhen; Liu, Li; Peng, Bo; Liu, Tianpeng; Li, Xiang

Online Non-Cooperative Radar Emitter Classification from Evolving and Imbalanced Pulse Streams

Published in:
IEEE Sensors Journal

DOI:
10.1109/JSEN.2020.2981976

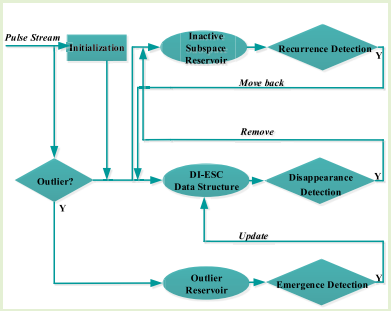
Published: 15/07/2020

Document Version
Publisher's PDF, also known as Version of record

Online Non-Cooperative Radar Emitter Classification From Evolving and Imbalanced Pulse Streams

Jinping Sui, Zhen Liu, Li Liu, Bo Peng, Tianpeng Liu, and Xiang Li

***Abstract*—Recent research treats radar emitter classification (REC) problems as typical closed-set classification problems, *i.e.,* assuming all radar emitters are cooperative and their pulses can be pre-obtained for training the classifiers. However, such overly ideal assumptions have made it difficult to fit real-world REC problems into such restricted models. In this paper, to achieve online REC in a more realistic way, we convert the online REC problem into dynamically performing subspace clustering on pulse streams. Meanwhile, the pulse streams have evolving and imbalanced properties which are mainly caused by the existence of the non-cooperative emitters. Specifically, a novel data stream clustering (DSC) algorithm, called dynamic improved exemplar-based subspace clustering (DI-ESC), is proposed, which consists of two phases, *i.e.,* initialization and online clustering. First, to achieve subspace clustering on subspace-imbalanced data, a static clustering approach called the improved ESC algorithm (I-ESC) is proposed. Second, based on the subspace clustering results obtained, DI-ESC can process the pulse stream in real-time and can further detect the emitter evolution by the proposed evolution detection strategy. The typically dynamic behavior of emitters such as appearing, disappearing and recurring can be detected and adapted by the DI-ESC. Extinct experiments on real-world emitter data show the sensitivity, effectiveness, and superiority of the proposed I-ESC and DI-ESC algorithms.**



***Index Terms*— Radar emitter classification, data stream clustering, imbalanced data stream, subspace clustering.**

I. Introduction

RADAR emitter classification (REC) based on the passively received radar pulse streams is of great importance in both military and civil systems, especially in electronic support measurement (ESM) systems [1]. The received radar pulse streams usually consist of sequences of pulses emitted from multiple radar transmitters [2]. Nowadays, the REC is becoming much more challenging than ever before as the electromagnetic environment gets unprecedentedly congested [3].

A conventional but currently ineffective method for REC is classifying the pulses according to their pulse description words (PDWs) which consist of pulse parameters like angle-of-arrival (AOA), carrier frequency *etc* [4]. However, with the wide application of complex modulation radars, traditional PDWs are not sufficient to describe modern complex radar pulses. Such PDW-based methods can no longer meet the requirements of modern REC tasks. Currently, the researchers focus on solving the REC problem through supervised learning approaches. Namely, they assume that the pulse samples of all radar emitters can be pre-obtained and used to train delicate classifiers for REC purposes. As a result, recent published REC work mainly focuses on two aspects, feature extraction and classifier selection. For feature extraction, various time-frequency (T-F) transform approaches, like short-time Fourier transform (STFT) [5], wavelet [6], quadratic T-F [7], Zhang *et al.* [8] and variational mode decomposition (VMD) [9] are adopted to extract more distinguishable intra-pulse features including intentional or even unintentional modulation features [9], [10]. Currently, with the introduction of more complex modulation techniques, the dimensions of feature vectors continue to increase. The ever-increasing dimensions

Manuscript received February 20, 2020; accepted March 16, 2020. Date of publication March 19, 2020; date of current version June 18, 2020. This work was supported by the National Natural Science Foundation of China under Grant 61701510 and Grant 61801488. The associate editor coordinating the review of this article and approving it for publication was Dr. Michael Antoniou. *(Corresponding author: Zhen Liu.)*

Jinping Sui is with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China, and also with the Department of Computer Science, Aalto University, 02150 Espoo, Finland (e-mail: suijinping13@nudt.edu.cn).

Zhen Liu, Bo Peng, and Tianpeng Liu are with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: zhen_liu@nudt.edu.cn; pengbo06@gmail.com; everliutianpeng@sina.cn).

Li Liu is with the College of System Engineering, National University of Defense Technology, Changsha 410073, China, and also with the Center for Machine Vision and Signal Analysis, University of Oulu, 90014 Oulu, Finland (e-mail: li.liu@oulu.fi).

Xiang Li is with the National University of Defense Technology (NUDT), Changsha 410073, China (e-mail: lixiang01@vip.sina.com).

Digital Object Identifier 10.1109/JSEN.2020.2981976

of feature vectors lead to more and more classifiers, such as support vector machine (SVM) [1], [7], [8], [11], relevance vector machine (RVM) [12], neural networks [1], [13], and even deep learning networks (*e.g.,* convolutional neural networks (CNN) [14] and recurrent neural networks (RNN) [4]) are introduced to classify the pulses based on the features.

A. Motivations

Despite that the recent work has promoted the research of REC to some extent, they have been found unrealistic in many real-world REC scenarios where vast non-cooperative emitters exist. Essentially, these recent REC works treats the REC problem as a typical closed-set classification problem, *i.e.,* the classes of the training and testing sets are identical, while ignoring three unique challenges existing in real-world REC tasks. First, as the pulse streams are continuously received, the REC tasks are expected to be addressed in real-time manners. However, the existing REC work costs considerable time and resources in samples labeling and classifiers training. Second, the pulse samples from non-cooperative radar emitters are scarce or even impossible to pre-obtain, causing the training set to be highly imbalanced. Thus, the classifiers trained based on such training sets will inevitably fail to achieve REC goals. Third, the behavior of radar emitters, especially the non-cooperative ones, is generally highly dynamic leading to the REC tasks cannot fit into the closed-set classification model. For example, there will be various non-cooperative emitters that emit pulses from a certain time, and the samples of these emitters are not in the training set at all. Additionally, some emitters only work within a certain period of time or some non-adjacent time periods. Such dynamic behavior usually contains significant information indicating the change underlying the working context and should be extracted timely by the REC algorithms to provide a deeper perception of the electromagnetic environment. Consequently, it is obviously unwise to continue using closed-set classification models to address real-life REC tasks.

B. Contributions

In essence, the radar pulse stream is a kind of data stream which has the following unique properties compared to general data streams, *i.e.,* unlabelled, high-dimensional, evolving and imbalanced. As discussed before, the evolving property is mainly caused by the dynamic behavior of the emitters and the imbalanced property is due to the pulses of the non-cooperative emitters are much more scarce. Despite the pulses are high-dimensional, it has been found that the radar pulses from the same emitters actually lie on or near low-dimensional subspaces [15], [16]. From this sense, the REC task can be deemed to perform an online and incremental subspace clustering on the pulse stream whose goal is to group the pulses from the same emitters together and separate the pulses from the different emitters. Such an online and incremental subspace clustering task can be achieved by the data stream subspace clustering technique which is a vital sub-topic of data stream clustering (DSC) research. However, performing data stream subspace clustering algorithms on evolving and/or imbalanced data streams has not been well addressed. Recently, an algorithm called exemplar-based subspace clustering (ESC) [17] is proposed to address the subspace-imbalanced problem, *i.e.,* the distribution of the data in each subspace is imbalanced. However, the ESC suffers from severe performance instability and is limited to dealing with static data sets instead of data streams.

In this paper, a data stream subspace clustering algorithm, called dynamic improved ESC (DI-ESC) algorithm, is proposed to achieve online REC on pulse streams, where the pulses from the same radar emitters are assumed being located in the same subspace. To the best of the author's knowledge, we are among the first researchers to consider subspace-imbalanced and evolving problems into designing DSC algorithms and to solve the REC problem in the DSC framework. In specific, we first propose the improved ESC (I-ESC) algorithm to overcome the performance instability of the original ESC algorithm. Different from the ESC algorithm randomly selecting partial point as the first exemplar, the proposed I-ESC algorithm ensures its stable and good performance by selecting the point with the smallest sum of similarities with all the other points as the initial exemplar. Based on the I-ESC algorithm, the DI-ESC is then proposed. DI-ESC consists of two phases, *i.e.,* initialization and online clustering. First, the proposed I-ESC algorithm is adopted in the first phase to obtain an initial clustering result. Then, based on the self-expressiveness property, the arriving data point in the second phase can be assigned in an online manner. A subspace evolution detection strategy is also be proposed to ensure the evolving behavior of subspaces, such as emerging, disappearing, and recurring can be detected and adapted. Compared with the state-of-the-art DSC algorithms, DI-ESC can perform DSC task towards imbalanced and evolving data streams. Experiments on real-world data streams collected in multi-emitter scenario prove the validity and superiority of DI-ESC compared with state-of-the-art DSC algorithms including scalable sparse subspace clustering (SSSC) [18], scalable low-rank representation (SLRR) [18], scalable least squares regression (SLSR) [18], clustering of evolving data streams into arbitrary shapes (CEDAS) [19] and stream affinity propagation (STRAP) [20].

C. Organizations

The remainder of this paper is organized as follows. Section II formulates the REC problem with the emitter-evolving and emitter-imbalanced property, and briefly introduce the original ESC algorithm. Section III presents the principles and methodology of the proposed algorithms. In Section IV, we verify and analyze the performance of the proposed algorithms by comparing with the state-of-the-art algorithms. Finally, the study is concluded in Section V.

II. Problem Formulation and Preliminaries

A. The Online REC Problem Formulation

Consider a typical REC scenario where l^t radar emitters (we call them emitters for brevity) work at timestamp t simultaneously. We denote each emitter as $\mathcal{E}$, and at t, the l^t emitters can be denoted as $\mathbb{E}^t = \{\mathcal{E}_l^t\}_{l=1}^{l^t}$. Then, for a receiver working in the same scenario, it would receive the pulse stream consisting of pulses continuously produced by the

emitters [9], [21]. Assume at each timestamp t, it receive one pulse $\boldsymbol{p}^t$ which is a D-dimensional vector, $\boldsymbol{p}^t \in \mathbb{R}^{D\times 1}$. Thus, the received pulse stream can be denoted as $\mathbf{P}=\{\boldsymbol{p}^t\}_{t=1}^{N}$ ($N \to \infty$). For ease of discussion, we use a matrix $\mathbf{P}^t$ to denote the pulses which we have received up to t, that is, $\mathbf{P}^t = [\boldsymbol{p}^1 \dots \boldsymbol{p}^t]_{D\times t}$. Note that we assume that there is no overlap among the pulses.

As discussed before, the pulse stream typically has the following unique properties: unlabelled, high-dimensional, emitter-evolving and emitter-imbalanced. The first two properties can be easily understood. Now we focus on further formulating the rest properties aforementioned.

1) The Emitter-Evolving Property: It refers to the dynamic behavior of emitters for different functions or in different working time, i.e., switching from power on/off to off/on.

Specifically, we mainly consider three types of emitter-evolving, *i.e.,* emitter emergence, emitter disappearance and emitter recurrence, which can be formulated as follows:

- *Emitter emergence*. It refers to the occurrence of a new emitter at t. In particular, an emitter $\mathcal{E}$ emerges at timestamp t if $\mathcal{E} \notin \mathbb{E}^1 \cup \mathbb{E}^2 \cup \dots \cup \mathbb{E}^{t-1}$ and $\mathcal{E} \in \mathbb{E}^t$.
- *Emitter disappearance*. It is defined as a previously existed emitter that is not working during the recent period. Formally, an emitter $\mathcal{E}$ disappears if $\mathcal{E} \in \mathbb{E}^{t_0} \cap \mathbb{E}^{t_0+1} \cap \dots \cap \mathbb{E}^{t-1}$ and $\mathcal{E} \notin \mathbb{E}^t$, where $1 \le t_0 < t$.
- *Emitter recurrence*. It means the situation where a previously disappeared emitter recurs at t. Formally, an emitter $\mathcal{E}$ recurs at t if $\mathcal{E} \in \mathbb{E}^{t_1} \cap \mathbb{E}^{t_1+1} \cap \dots \cap \mathbb{E}^{t_2-1}$, $\mathcal{E} \notin \mathbb{E}^{t_2} \cup \mathbb{E}^{t_2+1} \cup \dots \cup \mathbb{E}^{t-1}$, and $\mathcal{E} \in \mathbb{E}^t$, where $1 \le t_1 < t_2 < t$.

2) The Emitter-Imbalanced Property: Various emitters differ greatly in operating time and pulse emission frequency, resulting in an imbalanced distribution of the numbers of pulses received from each emitter.

Assume during the period from t_1 to t_2 ($t_2 > t_1$), there exists $l_{(t_1,t_2)}$ emitters which emit $n_i (i = 1, 2, \dots, l_{(t_1,t_2)})$ pulses, respectively. Relatively, those emitters with large (small) numbers of pulses are referred to as *over-represented (under-represented) emitters*. Furthermore, We denote the emitters with the maximum and minimum numbers of pulses as $\mathcal{E}_{max}$ and $\mathcal{E}_{min}$, respectively. To quantify the degree of imbalance, we define the *imbalance ratio* as

$$k = \frac{n_{max}}{n_{min}}, \tag{1}$$

where n_{max} and n_{min} denote the numbers of pulses emitted by the $\mathcal{E}_{max}$ and $\mathcal{E}_{min}$, respectively. Currently, most of existing works concentrate on imbalance ratios larger than 4 [22].

The online REC problem can be formulated as follows:

3) Online REC Problem: Given the pulse stream $\mathbf{P}^t$, the goal of online REC is to determine emitters $\mathbb{E}^t = \{\mathcal{E}_l^t\}_{l=1}^{l^t}$ at each timestamp t and associate each received pulse $\boldsymbol{p}^t$ with an emitter $\mathcal{E}_l$ ($l \in [1, l^t]$).

Essentially, the pulse stream is a unique kind of data stream with evolving and imbalanced properties. As discussed before, the pulses emitted by the same emitter are located on one subspace and the pulses emitted by the different emitters are on different subspaces. Therefore, the task of online REC can be converted into performing online subspace clustering on the data stream with subspace-evolving and subspace-imbalanced properties. Currently, in the field of subspace clustering, the self-expressiveness property of data is widely adopted by most of the state-of-the-art algorithms [17], [18], [23].

4) Self-Expressiveness Property: Assume there are N data points $\mathcal{Y} = [\boldsymbol{y}_1, \dots, \boldsymbol{y}_N]$ which are located in a union of subspaces. Each data point $\boldsymbol{y}_i$ is a D-dimensional vector. For each data point $\boldsymbol{y}_i$, there exists a linear combination of other points which can fulfill

$$\boldsymbol{y}_i = \mathcal{Y}\boldsymbol{r}_i, \tag{2}$$

where $\boldsymbol{r}_i = [r_{i1}, \dots, r_{iN}]^\top$ and $r_{ii} = 0$ to avoid the trivial solution of $\boldsymbol{y}_i = \boldsymbol{y}_i$. The $\boldsymbol{r}_i$ is referred to as the *representative coefficients*. Note that Eq. (2) has infinite solutions. However, it has been observed that if we restrict $\boldsymbol{r}_i$ to be sparse, we will obtain the $\boldsymbol{r}_i^*$ which has subspace-preserving property, that is, the $r_{ij} (j \in \{1, 2, \dots, N\})$ is nonzero only if $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are from the same subspace [17], [24], [25]. Such a subspace-preserving $\boldsymbol{r}_i^*$ can be obtained by the following sparse optimization problem

$$\min_{\boldsymbol{r}_i \in \mathbb{R}^N} \|\boldsymbol{r}_i\|_1 + \frac{\lambda}{2} \|\boldsymbol{y}_i - \sum_{i\neq j} r_{ij}\boldsymbol{y}_j\|_2^2, \tag{3}$$

where $\lambda > 0$ is an input parameter, $\|\cdot\|_1$ and $\|\cdot\|_2$ denote to the ℓ_1-norm and ℓ_2-norm, respectively.

B. Exemplar-Based Subspace Clustering (ESC) Algorithm

In reality, such a subspace-preserving $\boldsymbol{r}_i$ can be obtained usually when $\mathcal{Y}$ is subspace-balanced, *i.e.,* the numbers of data points from each subspace are relatively balanced in $\mathcal{Y}$. However, when $\mathcal{Y}$ is subspace-imbalanced, it can be observed the representative coefficients of points from under-represented subspaces are more likely to have nonzero entries corresponding to data points in over-represented subspaces, *i.e,* the points from under-represented subspaces are easily swallowed by the over-represented subspaces. Thus, it is difficult to find those under-represented subspaces in this case. Interestingly, this phenomenon has also been found in [17]. The authors propose the ESC algorithm to address this problem in performing subspace clustering on imbalanced data sets.

To address the subspace-imbalanced problem, ESC reduces the degree of imbalance by finding a subset $\mathcal{Y}_0^* \subseteq \mathcal{Y}$. Specifically, $\mathcal{Y}_0^*$ is obtained by minimizing a self-representation cost function $F_\lambda(\mathcal{Y}_0)$, *i.e.,*

$$\mathcal{Y}_0^* = \arg\min_{|\mathcal{Y}_0|\le N_0} F_\lambda(\mathcal{Y}_0), \tag{4}$$

where

$$F_\lambda(\mathcal{Y}_0) = \sup_{\boldsymbol{y}_i \in \mathcal{Y}} f_\lambda(\boldsymbol{y}_i, \mathcal{Y}_0), \tag{5}$$

and

$$f_\lambda(\boldsymbol{y}_i, \mathcal{Y}_0) = \min_{\boldsymbol{r}_i \in \mathbb{R}^N} \|\boldsymbol{r}_i\|_1 + \frac{\lambda}{2} \| \boldsymbol{y}_i - \sum_{j:\boldsymbol{y}_j \in \boldsymbol{\mathcal{Y}_0}} r_{ij}\boldsymbol{y}_j\|_2^2 . \tag{6}$$

The points in $\mathcal{Y}_0^*$ are called *exemplars* and $\mathcal{Y}_0^*$ is referred to as *exemplar set*. N_0 is the number of the exemplars in

the exemplar set and needs to be specified as a prior by the users. Note that Eq. (6) is NP-hard in general and it can be approximately addressed by the farthest first search (FFS) algorithm proposed in [17]. Once $\mathcal{Y}_0^*$ is obtained, the $\boldsymbol{r}_i$ of each data point $\boldsymbol{y}_i$ in $\mathcal{Y}$ could also be obtained by Eq. (4) - Eq. (6). The $\boldsymbol{r}_i$ could be employed to build an affinity graph and the clustering result is obtained after applying spectral clustering to the affinity graph. However, ESC still suffers from two severe limitations: i) The performance of ESC is extremely unstable due to the introduction of a random selection in the FFS algorithm (more details will be given in Section III). ii) ESC can only process static data sets instead of data streams so far. The limitations mentioned above result in that ESC is not capable of processing data streams with subspace-imbalanced and subspace-evolving properties.

III. Improved ESC Algorithm and Dynamic Improved ESC Algorithm

The emitter-evolving and emitter-imbalanced properties of pulse streams pose great challenges for online REC achieving. In this section, we convert the online REC problem into an online subspace clustering problem on data streams with subspace-evolving and subspace-imbalanced properties, where each emitter is considered as a subspace. Specifically, we first propose a static subspace clustering algorithm, called improved ESC (I-ESC). Compared with original ESC, the proposed I-ESC achieves more robust subspace clustering performance on subspace-imbalanced data sets. Then, based on the proposed I-ESC, the dynamic I-ESC (DI-ESC) algorithm is further proposed for processing data streams with subspace-evolving and subspace-imbalanced properties.

Note that in addition to the online REC problem, DI-ESC can address similar data stream online subspace clustering problems in other fields. Therefore, in the following, we no longer emphasize the REC background to make DI-ESC a universal model that can be used in other fields. The pulse stream and the emitter are abstracted as a data stream and a subspace, respectively.

A. Improved ESC Algorithm

Essentially, ESC significantly reduces the imbalance degree of the original data set $\mathcal{Y}$ by identifying the exemplar set (*i.e.,* $\mathcal{Y}_0^*$) using Eq. (4) - Eq. (6). According to [17], the approximate solution of Eq. (4) can be obtained by the FFS algorithm. Concretely, FFS randomly selects a point from the data set $\mathcal{Y}$ as the first exemplar, denoted as $\mathcal{Y}_0^{(1)}$), and then progressively identifies other exemplars to form the exemplar set based on the $\mathcal{Y}_0^{(1)}$). However, such a random selection of the first exemplar causes the ESC algorithm to be extremely sensitive to $\mathcal{Y}_0^{(1)}$) and thus the performance of ESC is unstable. Obviously, the selection of the first exemplars is quite important for the determination of $\mathcal{Y}_0^*$ and it should not be selected randomly.

$F_\lambda(\mathcal{Y}_0)$ can be minimized when $\mathcal{Y}_0$ contains more data points which cannot be represented by the remaining points with small errors [17]. Namely, those points which are not similar to the rest tend to be selected as exemplars under the constraints of the optimization function in Eq. (4) - Eq. (6).

Algorithm 1 The Improved ESC Algorithm

Input: Data $\mathcal{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N] \subseteq \mathbb{R}^{D\times N}$, parameters $\lambda > 1$, η, p.

Output: The subspace clustering result of $\mathcal{Y}$.

1: Identify the exemplar set $\mathcal{Y}_0^*$ by Eq. (4) - Eq. (9).
2: Compute the sparse representative coefficients $\{\boldsymbol{r}_i\}_{i=1}^N$ by Eq. (10);
3: Normalize the $\{\boldsymbol{r}_i\}_{i=1}^N$ by $\widehat{\boldsymbol{r}}_i = \boldsymbol{r}_i / \| \boldsymbol{r}_i \|_2$;
4: Build the affinity matrix $\mathbf{W} = \mathbf{A} + \mathbf{A}^\top$, where $\mathbf{A}_{ij} = 1$ if $\widehat{\boldsymbol{r}}_j$ is a p-nearest neighbor of $\widehat{\boldsymbol{r}}_i$ and 0 otherwise;
5: Apply a spectral clustering [26], [27] algorithm to $\mathbf{W}$ aiming at obtaining the subspace clustering result on $\mathcal{Y}$.

Here, we propose a direction to identify the first exemplar, *i.e.,*

$$\mathcal{Y}_0^{(1)*} = \min_{\boldsymbol{y}_i \in \mathcal{Y}} \left[\sum_{\substack{\boldsymbol{y}_j \in \mathcal{Y} \\ j \neq i}} S(\boldsymbol{y}_i, \boldsymbol{y}_j) \right] \tag{7}$$

where $S(\cdot)$ is a similarity function. For example, $S(\cdot)$ can be defined as the opposite of Euclidean distance, correlation or any function that can measure the similarity of two data points. Considering the limitation of Euclidean distance in high-dimensional space, we define $S(\cdot)$ as the correlation function in this paper, that is,

$$S(\boldsymbol{y}_i, \boldsymbol{y}_j) = \frac{Cov(\boldsymbol{y}_i, \boldsymbol{y}_j)}{\sqrt{\boldsymbol{y}_i}\sqrt{\boldsymbol{y}_j}} = \frac{E(\boldsymbol{y}_i - E(\boldsymbol{y}_i))E(\boldsymbol{y}_j - E(\boldsymbol{y}_j))}{\sqrt{\boldsymbol{y}_i}\sqrt{\boldsymbol{y}_j}} \tag{8}$$

where $E(\cdot)$ is the expectation function.

Based on the $\mathcal{Y}_0^{(1)*}$ selected by Eq. (7), the FFS can progressively identify the exemplars using the following function.

$$\mathcal{Y}_0^{(i+1)*} = \mathcal{Y}_0^{(i)*} \cup \arg\max_{\boldsymbol{y}\in\mathcal{Y}} f_\lambda(\boldsymbol{y}, \mathcal{Y}_0^{(i)}) \tag{9}$$

Different from ESC, I-ESC introduces a new parameter η to control the size of the exemplar set $\mathcal{Y}_0$ instead of requiring the user to input a specific size N_0 directly. That is, $N_0 = \lceil N * \eta \rceil$ where $\lceil \cdot \rceil$ represents an integer that is the closest to and greater than $N * \eta$. After $\mathcal{Y}_0$ being identified, the subspace clustering could be then obtained based on $\mathcal{Y}_0^*$. Specifically, for each $\boldsymbol{y}_i \in \mathbf{Y}$ $(i = 1, 2, \ldots, N)$, we obtain its sparse representative coefficients $\boldsymbol{r}_i$, under the exemplar subset $\mathcal{Y}_0^*$, that is,

$$\min_{\boldsymbol{y}_i \in \mathbb{R}^D} \|\boldsymbol{y}_i\|_1 + \frac{\lambda}{2} \|\boldsymbol{y}_i - \sum_{j:\boldsymbol{y}_j \in \mathcal{Y}_0^*} r_{ji}\boldsymbol{y}_j\|_2^2 \tag{10}$$

After obtaining the sparse representative coefficients for each point in $\mathcal{Y}$, that is $\{\boldsymbol{r}_i\}_{i=1}^N$, the nearest neighbor approach is utilized to achieve the subspace clustering on $\mathcal{Y}$ (See **Algorithm 1**).

B. Online REC by Dynamic Improved ESC (DI-ESC) Algorithm

Compared with ESC, I-ESC achieves subspace clustering on subspace-imbalanced data sets with more robust performance. However, the application of I-ESC is still restricted to static data sets processing instead of data streams processing.

Therefore, in this section, we aim at extending I-ESC to an online algorithm, called dynamic I-ESC (DI-ESC), which can perform online subspace clustering on data streams with subspace-evolving and subspace-imbalanced properties. Three main difficulties exist when extending I-ESC to DI-ESC. i) DSC tasks have stricter limits on computing speed and storage space, which results in that DI-ESC cannot save all the data points. Therefore, DI-ESC is expected to only save the general information of the data streams instead of all the original data points. ii) The processing manner of I-ESC should be changed properly from batch processing to online processing. iii) The subspace-evolving property of data streams needs to be detected and adapted efficiently. Accordingly, in Section III-B1, a data structure, named *DI-ESC data structure*, is proposed to record the statistic summaries of the data streams. Then, in Section III-B2, we explained how DI-ESC achieves online subspace clustering on data streams based on the DI-ESC data structure. Lastly, a detection strategy is proposed in Section III-B3 to ensure DI-ESC is competent to adapt to the subspace-evolving property.

1) The DI-ESC Data Structure: The evolving property of data streams requires DI-ESC to accordingly provide dynamic subspace clustering results. Therefore, the DI-ESC data structure, denoted as $\mathbb{S}^t$, is proposed to ensure DI-ESC being facile to reflect the current patterns of the data streams. To avoid saving disappeared subspace in $\mathbb{S}^t$, at each timestamp, the state of each found subspace is evaluated, *i.e.,* being active or inactive. Here, 'inactive' means the corresponding subspaces are expired, *i.e.,* there are data points being assigned into these subspaces during the recent time period. The active state and the inactive state can convert to each other over time. It should be noted that only the information of active subspaces is reserved in $\mathbb{S}^t$. The information of the inactive subspaces is reserved into *inactive subspaces reservoir*, denoted as $\mathbb{D}^t$. Assume that at t, there are l^t active subspaces and h^t inactive subspaces. Then $\mathbb{S}^t = \{\mathcal{S}_i^t\}_{i=1}^{l^t}$ and $\mathbb{D}^t = \{\mathcal{D}_i^t\}_{i=1}^{h^t}$, where $\mathcal{S}_i^t$ and $\mathcal{D}_i^t$ denote to the information of active and inactive subspaces, respectively. We refer to $\mathcal{S}_i^t$ or $\mathcal{D}_i^t$ as the *subspace summaries*. Precisely, $\mathcal{S}_i^t = \{n_i^t, \mathbf{R}_i^t, \psi_i, p_i^t, q^t\}$ and $\mathcal{D}_i^t = [\widetilde{n}_i^t, \widetilde{\mathbf{R}}_i^t, \widetilde{\psi}_i, \widetilde{p}_i^t, \widetilde{q}^t]$, where

- n_i^t (respectively, $\widetilde{n}_i^t$) is the total number of data points assigned to active (resp. inactive) subspace i up to t;
- $\mathbf{R}_i^t$ ($\widetilde{\mathbf{R}}_i^t$) is called the *reserved data matrix* of active (inactive) subspace i which saves exemplars from subspace i up to t;
- ψ_i ($\widetilde{\psi}_i$) is the last timestamp when a point is assigned to subspace i;
- p_i^t ($\widetilde{p}_i^t$) is a scalar whose initial value is set as 0;
- q^t ($\widetilde{q}^t$) is the total number of outliers found in the data stream up to t. It should be noted that for all active and inactive subspaces, $q^t = \widetilde{q}^t$.

2) The Initialization and Online Clustering: The DI-ESC algorithm can be divided into two phases, *i.e.,* initialization and online clustering. In the first phase, the I-ESC algorithm (*i.e.,* **Algorithm 1**) is adopted to process the first batch of T_0 arriving data points, *i.e.,* $\mathbf{P}^{T_0}$, and the clustering results are summarized and stored in the DI-ESC data structure, *i.e.,* $\mathbb{S}^{T_0}$. Assume that $\mathcal{P}_0$ is the exemplar set selected from $\mathbf{P}^{T_0}$ by I-ESC algorithm, then $\mathbf{R}_i^{T_0}$ is a matrix consisting of the points belonging to subspace i in $\mathcal{P}_0$. $n_i^{T_0}$ is initialized by the number of points assigned into subspace i in $\mathbf{P}^{T_0}$. Therefore, for the subspace i found in the initialization phase, its subspace summary is $\mathcal{S}_i^{T_0} = \{n_i^{T_0}, \mathbf{R}_i^{T_0}, T_0, 0, 0\}$. It is noted that $\mathbb{D}^{T_0} = \emptyset$ in the first phase.

In the second phase, DI-ESC then progressively processes each arriving data point $\boldsymbol{p}^t$ ($t \geq T_0$) in an online manner. For each $\boldsymbol{p}^t$, it can be either from a subspace (active subspace or inactive subspace) that has been found or an upcoming new subspace that needs to be found. Accordingly, the data points from the found subspaces are called *normal points*. Otherwise, they are called *outliers*. The proposed DI-ESC algorithm needs to identify that $\boldsymbol{p}^t$ is an outlier or not firstly. Here, we define a matrix $\mathbf{Z}^t = [\mathbf{R}_1^t \cdots \mathbf{R}_{l^t}^t \, \widetilde{\mathbf{R}}_1^t \cdots \widetilde{\mathbf{R}}_{h^t}^t]$ consisting of all reserved data matrix of active and inactive subspaces at t. Each reserved data matrix in $\mathbf{Z}^t$ is actually a sub-block of $\mathbf{Z}^t$.

Similar to Eq. (2), the representative coefficients $\boldsymbol{r}^{*t}$ of each data point $\boldsymbol{p}^t$, can be obtained by Eq. (3). For normal points, $\boldsymbol{r}^{*t}$ has the subspace-preserving property. That is, non-zero elements of $\boldsymbol{r}^{*t}$ are concentrated in a certain sub-block (this sub-block corresponds to the reserved matrix of its own subspace). While for outliers, the non-zero coefficients of $\boldsymbol{r}^{*t}$ do not have such property. Here we introduce the sparsity concentration index (SCI)[1] [28], [29] to quantitatively measure the concentration of non-zero coefficients of $\boldsymbol{r}^{*t}$.

According to [28], $\mathrm{SCI}(\boldsymbol{r}^{*t}) \in [0, 1]$ and a higher $\mathrm{SCI}(\boldsymbol{r}^{*t})$ means the coefficients of $\boldsymbol{r}^{*t}$ are more likely to concentrate in a single subspace. We further introduce a threshold τ and accept $\boldsymbol{p}^t$ as a normal point if

$$\mathrm{SCI}(\boldsymbol{r}^{*t}) \geqslant \tau, \tag{11}$$

and otherwise identify $\boldsymbol{p}^t$ as an *outlier*. The outliers will be stored in an outlier reservoir, denoted as $\mathcal{O}^t$. Note that q^t is the number of outliers in $\mathcal{O}^t$ and $q^t = \widetilde{q}^t$. For the normal data point, we further calculate the residual when assigning $\boldsymbol{p}^t$ to the j sub-block ($j = 1, 2, \ldots, l^{t-1} + h^{t-1}$). Then, the optimal sub-block to assign $\boldsymbol{p}^t$ can be obtained by the following optimization function,

$$\min_{j^*} \quad s_j(\boldsymbol{p}^t) = \| \boldsymbol{p}^t - \mathbf{Z}^{t-1}\delta_j(\boldsymbol{r}^{*t}) \|_2 . \tag{12}$$

where $\delta_j(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ is a function that selects the coefficients associated with the jth ($j \in [1, l^{t-1} + h^{t-1}]$) subpart in $\boldsymbol{r}^{*t}$ and keeps its elements which correspond to other subsparts in $\boldsymbol{r}^{*t}$ as zero.

After obtaining the subspace j^* to assign $\boldsymbol{p}^t$, the subspace summary of subspace j^* is accordingly updated. Concretely, for the active (inactive, respectively) subspace, $n_i^t = n_i^{t-1} + 1$ ($\widetilde{n}_i^t = \widetilde{n}_i^{t-1} + 1$), $\psi_i = t$ ($\widetilde{\psi}_i = t$), and $p_i^t = p_i^{t-1} + 1$ ($\widetilde{p}_i^t = \widetilde{p}_i^{t-1} + 1$).

[1]For a coefficient vector $\boldsymbol{x} \in \mathbb{R}^N$, $\mathrm{SCI}(\boldsymbol{x}) = (m \cdot \max_i(\| \delta_i(\boldsymbol{x}) \|_1 / \| \boldsymbol{x}^t \|_1) - 1)/(m-1)$, where m is the number of the sub-blocks of $\boldsymbol{x}$, $\delta_i : \mathbb{R}^N \to \mathbb{R}^N$ is the characteristic function that selects the coefficients associated with the ith sub-block.

Algorithm 2 DI-ESC Algorithm

Input: Data stream $\mathbf{P}$; Initial batch size T_0; Thresholds $\tau, \alpha, \beta, \gamma$, $\mathbb{D} \leftarrow \emptyset$, $\mathbb{O} \leftarrow \emptyset$

Output: The online clustering result of the data stream.

1: Apply I-ESC algorithm (**Algorithm 1**) to initialize DI-ESC data structure.
2: For each arriving point $\boldsymbol{p}^t$ ($t > T_0$), compute SCI($\boldsymbol{r}^{*t}$).
3: Determine if $\boldsymbol{p}^t$ is a normal or outlier via Eq. (11). For the normal point, compute Eq. (12) and then update $\mathbb{S}$ or $\mathbb{D}$ accordingly. For the outlier, update $\mathbb{O}$.
4: Emergence detection: apply I-ESC algorithm (**Algorithm 1**) to $\mathbb{O}$ if $q^t \geqslant \alpha$.
5: Disappearance detection: for each active subspace, compute Δ_i. Remove the $\mathcal{S}_i$ from $\mathbb{S}$ to $\mathbb{D}$ if $\Delta_i < 0.5$.
6: Recurrence detection: for each active subspace, compare $\widetilde{p}_i^t$ with γ. If $\widetilde{p}_i^t \geqslant \gamma$, then remove the $\mathcal{D}_i$ from $\mathbb{D}$ to $\mathbb{S}$.

3) Subspace Evolution Detection: The proposed DI-ESC algorithm can detect three typical types of subspace evolution *i.e.,* subspace emergence, disappearance and recurrence. When new subspaces start emerging, their data points actually will be identified as outliers by the proposed DI-ESC because there are no existing subspaces to assign these data points. Hence, at each timestamp, DI-ESC checks if q^t exceeds an emergence detection threshold α. If $q^t \geqslant \alpha$, the I-ESC (**Algorithm 1**) will be applied to the data matrix consisting of all points in $\mathcal{O}^t$ to find new subspaces and their subspace summaries. The subspace summaries of the new emerging subspaces will be added to the DI-ESC data structure.

It is possible for some active subspaces to gradually become inactive when fewer and fewer data points are assigned. We propose a decay function to quantitatively measure the active degree of active subspace i ($i \in [1, l^t]$) at each timestamp, denoted as Δ_i, *i.e.,*

$$\Delta_i = 1 - \frac{1}{1 + e^{-(t-\psi_i-\beta)}}. \tag{13}$$

where β is the maximum tolerant duration of the DI-ESC algorithm for the active subspaces that are not accessed by the points. The active subspace continues to be accepted as an active subspace only if its corresponding $\Delta_i \geq 0.5$. Namely, if there is no point being assigned into the subspace i during the recent β timestamps, the subspace i will be accepted as an inactive subspace rather an active subspace.

Additionally, some inactive subspaces are possible to become active again. For each inactive subspace, $\widetilde{p}_i^t$ will be compared with a threshold γ at each timestamp. An inactive subspace will be accepted as an active subspace if $\widetilde{p}_i^t \geqslant \gamma$.

IV. Numerical Experiments

In this section, we set up experiments to investigate the performance of the proposed I-ESC and DI-ESC algorithms in dealing with static subspace-imbalanced data sets and dynamic subspace-imbalanced data streams, respectively. The data sets and data streams are collected from the real-world REC scenario. Specifically, we first verify the I-ESC algorithm by comparing its performance with that of ESC algorithm on a real-world data set under different parameter settings. Second, the DI-ESC algorithm, as well as other baseline algorithms, are all applied to several imbalanced and evolving real-world data streams. Then, their performance is compared and analyzed. The sensitivity of I-ESC and DI-ESC is analyzed after their validation.

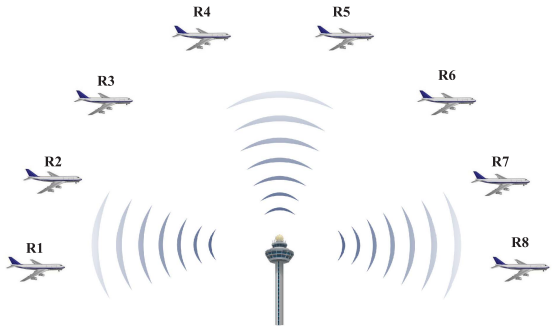


Fig. 1. The illustration of the operative scenario.

A. Data Sets, Baseline Algorithms and Evaluation Metrics

1) Data Sets: The data we collected is from the secondary surveillance radar(SSR) system. The SSR system has a wide range of applications, such as distinguishing fighters from ours in the military field, or air control in the civil aviation field. The SSR system mainly includes an interrogator equipped on the ground and a transponder equipped on the aircraft. It has been found that, due to aging, temperature, frequency stability drift, *etc.*, the transponder inevitably introduces UMOP (Unintentional Modulation of Pulses). Generally, it is assumed that UMOP caused by the same transponder is stable and consistent. Therefore, the pulses from the same transponder can be grouped together according to UMOP. Our data is the pulse emitted by different transponders (8 transponders in total, notated as R1 to R8). Fig. 1 illustrates the operative scenario. The data is collected at about 15km from eight different civil aircraft during their takeoff (separately collected). By analyzing the data, the signal to noise ratio (SNR) of the operative scenario is approximately 15dB (measured from the overall data). Precisely, the data collected from every single transponder is regarded as one subspace. Each data point is represented by a 400-dimension vector. The static datasets and dynamic data streams are further generated based on the real-world data.

2) Subspace-Imbalanced Datasets: Several static data sets with different imbalance ratios, *i.e.,* k, ranging from 4 to 10 are generated to study the superiority of the I-ESC algorithm compared to the ESC algorithm. The first four emitters (*e.g.,* R1-R4) are set as under-represented emitters and the rest (*e.g.,* R5-R8) are regarded as over-represented ones.

3) Subspace-Imbalanced and Evolving Data Streams: Beyond the validation of the proposed I-ESC algorithm, several subspace-imbalanced data streams (denoted as DS1 - DS4) with different evolving properties are additionally generated to test the effectiveness of the DI-ESC algorithm. The basic information of these tested data streams is summarized in Table I.

TABLE I
THE BASIC INFORMATION OF IMBALANCED DATA STREAMS TESTED IN THE EXPERIMENTS

Streams	evol. property	# initial subspaces	# subspaces
DS1	emerg.	4	8
DS2	emerg.	4	8
DS3	emerg.	2	8
DS4	disap. & recur.	8	8

4) Baseline Algorithms: For the first experiment, we compare the I-ESC only with ESC [17] because the purpose of this experiment is to demonstrate the effectiveness and superiority of the I-ESC in handling with imbalanced data sets. In the second experiment, we compare DI-ESC algorithm with D-ESC and other five state-of-the-art DSC algorithms, CEDAS [19], STRAP [20], SSSC [18], SLSR [18], and SLRR [18]. Note that the D-ESC algorithm is proposed here which is based on ESC algorithm. By comparing DI-ESC and D-ESC, we can further investigate the superiority of the I-ESC algorithm compared with ESC algorithm. CEDAS and STRAP are typical density-based and distance-based DSC methods, respectively. SSSC, SLSR, and SLRR are three representation-based subspace learning methods which are capable of handling high-dimensional data streams.

5) Evaluation Metrics: In this paper, the clustering quality of all algorithms is measured using accuracy and normalized mutual information (NMI) between the results given by the algorithms and the ground truth. The values of accuracy and NMI are real numbers between 0 and 1. Particularly, larger values mean the given result matches the ground truth more. In our experiments, the accuracy and NMI are the average values obtained by running each algorithm 50 times on each data set or data stream. We carried out all the experiments on a computer with 2.3GHz CPU and 4Gb memory.

B. The Validation of I-ESC

The I-ESC and ESC algorithms are applied on the imbalanced static data sets where the imbalance ratios range from 4 to 10. The corresponding results are depicted in Fig. 2, from which the following could be observed.

- I-ESC can effectively handle subspace-imbalanced data sets and its performance is significantly better than the ESC algorithm. Besides, the performance of I-ESC is very stable with variance closing to zero during 50 replicate experiments. Instead, the performance of ESC algorithm fluctuates relatively greater. This is mainly due to that ESC randomly selects a certain point as its initial exemplar and thus makes it difficult to maintain good and stable performance.
- The imbalance ratio of the data set has an impact on the performance of the I-ESC and ESC algorithms. Overall, as the imbalance ratio increases, the performance of the I-ESC and ESC algorithms will decrease accordingly. For example, when $k = 4$, the accuracy and NMI of I-ESC are 0.7390 and 0.7684 (for ESC, the corresponding results are 0.7210 and 0.7550). While when $k = 10$, the accuracy and NMI of I-ESC reduce to 0.6300 and 0.5336 (for ESC, the corresponding results are 0.5736 and 0.5081).

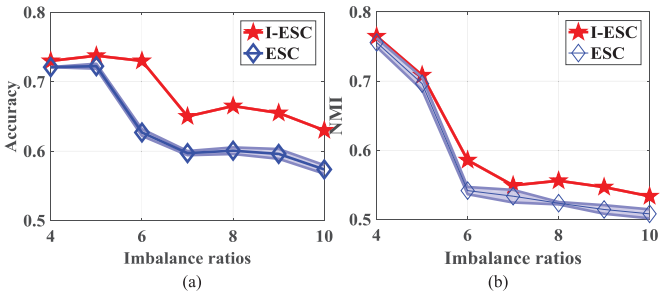


Fig. 2. The clustering quality, (a) accuracy and (b) NMI, of the I-ESC and ESC on imbalanced data sets with different imbalance ratios ranging from 4 to 10 ($\eta = 0.3$).

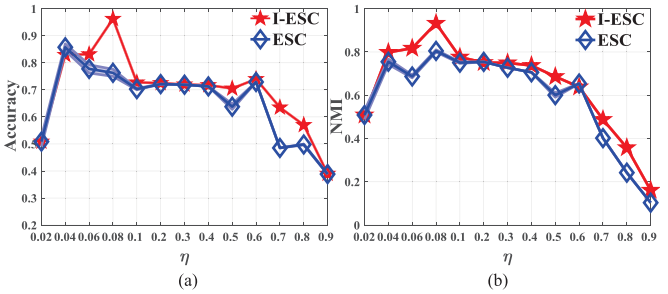


Fig. 3. The influence of paprameter η on the clustering quality, (a) accuracy and (b) NMI, of I-ESC and ESC.

In addition, to study the influence of the size of exemplar set which is controlled by the parameter η, we further perform experiments by setting different η on the imbalance data set whose $k = 4$. Fig. 3 reports the accuracy and NMI of the results. The experimental results are in line with our expectation. It can be observed that the performance of I-ESC, as well as ESC, shows downward trend with the increase of η in general, although there is slight fluctuation. This is mainly because as η increases, the imbalance of the exemplar set $\mathcal{Y}_0^*$ also begins to increase. When η is large enough, the $\mathcal{Y}_0^*$ is close to the original set $\mathcal{Y}$ which is quite imbalanced. However, it does not mean that η should be set extremely small because insufficient exemplars are not enough to represent the entire subspaces.

C. The Validation of DI-ESC

In the following experiments, we evaluate the performance of the DI-ESC method as well as the baseline algorithms on processing imbalanced data streams with different evolving properties.

1) The Data Streams With Emerging Property: We carry out experiments on three emerging data streams (DS1-DS3). The evolving properties of data streams are depicted in Fig. 4(a) - Fig. 4(c), respectively. In Fig. 4, the x-axis is the timestamp of the corresponding stream and the y-axis is the different subspaces from 1 to 8. It can be observed that these data streams are imbalanced as a result of the data points are not equally distributed among the subspaces. Meanwhile, these data streams have different emerging properties. For instance, DS1 has 4 subspaces at the initial stage and 4 new emerging subspaces at the second stage, while DS3 only has 2 subspaces at the first stage but 6 new emerging subspaces in the following stage.

TABLE II
PERFORMANCE COMPARISON DIFFERENT ALGORITHMS OVER THREE EMERGING DATA STREAMS (DS1-DS3)

Data stream	DS1			DS2			DS3		
Algorithm	Acc.(%)	NMI(%)	Time(s)	Acc.(%)	NMI(%)	Time(s)	Acc.(%)	NMI(%)	Time(s)
DI-ESC	**77.14**	**83.62**	1.36	**80.49**	**81.46**	1.48	**76.88**	**78.98**	1.72
D-ESC	67.60	75.93	1.24	74.57	80.30	1.18	75.73	77.85	1.54
SSSC [19]	58.47	43.48	11.03	59.02	43.48	10.01	25.00	16.48	7.24
SLRR [19]	59.02	43.48	13.07	59.02	43.48	10.78	25.00	16.48	3.07
SLSR [19]	59.03	43.86	**0.52**	59.03	43.86	**0.62**	25.00	16.48	**0.53**
CEDAS [20]	26.81	41.08	1.98	26.39	41.49	2.01	27.08	42.82	2.07
STRAP [21]	26.25	36.17	1.11	26.94	36.43	1.16	28.61	34.59	0.78

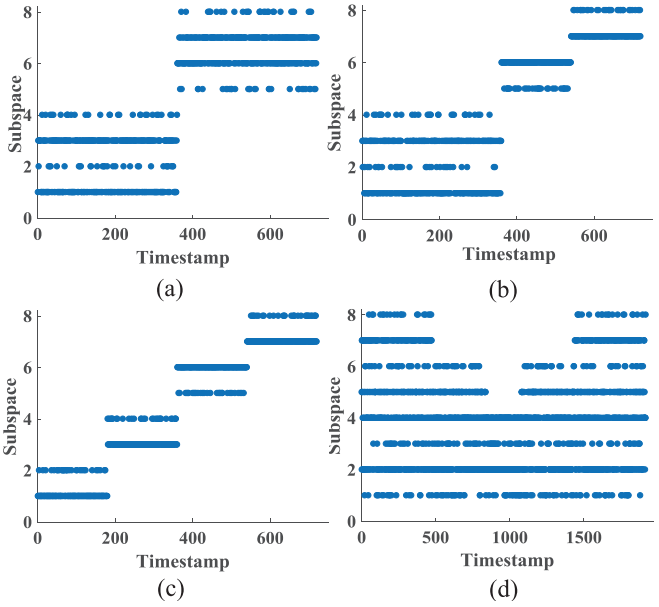


Fig. 4. The evolving properties of (a) DS1. (b) DS2. (c) DS3. (d) DS4.

Table II reports the processing results of DI-ESC and other baseline algorithms for these data streams, from which we have the following observations.

- DI-ESC and D-ESC are both effective in handling with data streams with emerging and imbalanced properties and they outperforms other algorithms in accuracy and NMI by a considerable performance margin. For instance, for DS1, the performance of the best algorithm (SLSR) except DI-ESC and D-ESC is 0.5903 in accuracy and 0.4386 in NMI. While DI-ESC and D-ESC achieve 0.7714 as well as 0.6760 in accuracy and 0.8362 as well as 0.7593 in NMI. The reason behind this is that the SSSC, SLRR, and SLSR are not capable of detecting evolving property of the data streams and they just assume that the subspace structures are stationary. As for CEDAS and STRAP, they find the subspaces based on the Euclidean distance which is not effective in high-dimensional feature space.
- Except for the SLSR algorithm, the rest of the algorithms are not as efficient as DI-ESC and D-ESC algorithms due to more computational time required. Generally, SSSC consumes the longest time because it requires initialization using the SSC [24] algorithm which is inefficient. Although DI-ESC and D-ESC also adopt I-ESC and ESC to initialize the models respectively, they select some exemplars to participate in the initial subspace finding and greatly save the computational time.
- It can be observed that DI-ESC outperforms D-ESC in accuracy and NMI, which can be attributed to the fact that I-ESC achieves a better performance than ESC. DI-ESC consumes slightly more time than D-ESC because of the introduction of Eq. (7) and Eq. (8) in I-ESC.

2) The Data Streams With Disappearing and Recurring Properties: In this section, we further investigate the performance of the proposed DI-ESC on the disappearing and recurring imbalanced data stream, *i.e.,* DS4. The evolving property of DS4 has been depicted in Fig. 4(d). It can be observed from Fig. 4(d) that the distribution of the points among the 8 subspaces is imbalanced. Additionally, some subspaces are not visited by the arriving points during the period approximately from $t = 500$ to $t = 1400$. Namely, these subspaces are disappeared during this period. However, at the end of the data stream, the disappeared subspaces become active again (subspace recurrence).

We carry out experiments on DS4 to show the effectiveness and superiority of the proposed method on detecting the disappearing and recurring subspaces by comparing with the baseline algorithms. Fig. 5 shows the corresponding results. As can be observed in Fig. 5(a), DI-ESC successfully detects the evolving subspace structure underlying DS4. Particularly, DI-ESC tracks the disappearance of 4 subspaces firstly and then detects the recurrence of these disappeared subspaces. SSSC, SLRR, SLSR (see Fig. 5(b)) and STRAP (see Fig. 5(d)) can not detect any subspace disappearance and recurrence. This is because that these models are based on the assumption that the subspace structures of the data streams keep stationary. Even though CEDAS can track the subspace evolution in theory, its performance on high-dimensional data streams, as shown in Fig. 5(c), cannot be ensured due to relying on traditional distance-based similarity measurement.

D. The Sensitivity Analysis of DI-ESC

The key parameters η, τ and α have great impacts on the performance of DI-ESC. In particular, the η parameter indirectly affects the performance of the DI-ESC algorithm by affecting I-ESC, which has been discussed in Section IV-B. In this section, we focus on analyzing the effects of τ and α.

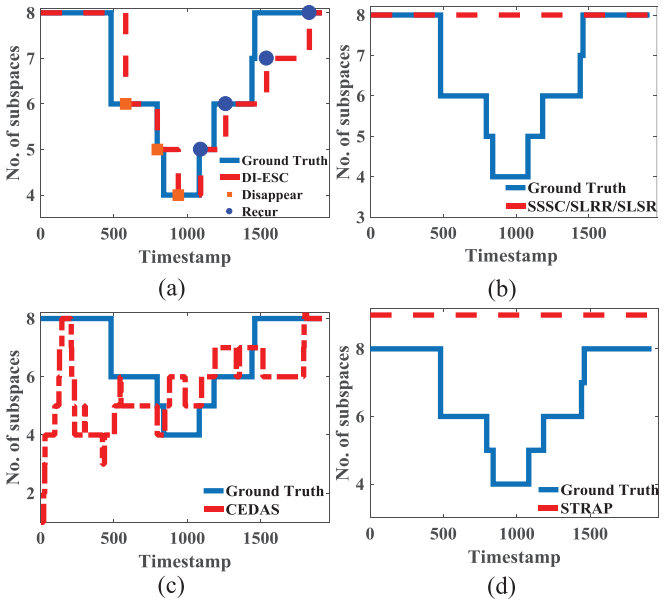


Fig. 5. The real-time number of the subspace recovered by algorithms on data stream 4. (a) DI-ESC, (b) SSSC/SLSR/SLRR, (c) CEDAS, (d) STRAP.

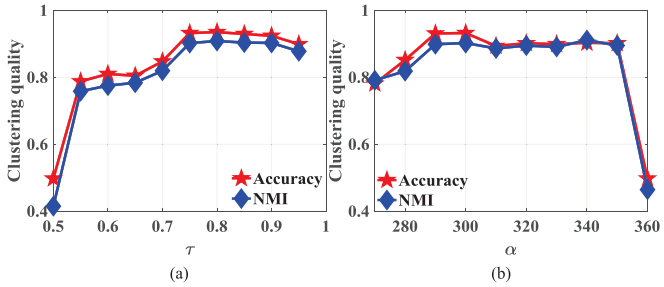


Fig. 6. The parameter sensitivity of DI-ESC on parameters. (a) Influence of τ. (b) Influence of α.

τ and α control the sensitivity of the model to the outliers and emergence of new subspaces, respectively.

To study the influences of the τ and α, we perform experiments on the DS1 by setting different parameters. The corresponding results are illustrated in Fig. 6, from which the following observations can be obtained.

- Fig. 6(a) depicts the clustering quality of DI-ESC under different τ settings varying from 0.5 to 0.95 ($\alpha = 300$). When τ is relatively small, the clustering quality is low because outliers are more likely to be wrongly identified as normal points. This leads to a negative impact on the following process. However, it does not mean that τ should be set very large because a larger τ causes more normal points to be identified as outliers. For example, as illustrated in Fig. 6(a), when $\tau > 0.8$, the clustering quality starts decreasing.
- The influence of α on clustering quality is shown in Fig. 6(b), from which we can observe that the clustering quality is much lower when α is relatively small or large. Essentially, α determines the sensitivity of the DI-ESC model to the emergence of new subspaces, which affects the stability of the model. A small α will cause the emergence detection to be easily triggered, resulting in instability of the model. A large α is likely to cause the model to be insensitive to new subspaces, which seriously affects the clustering quality.

V. Conclusion

In this paper, we convert REC problems into performing subspace clustering on pulse streams which have subspace-evolving and subspace-imbalanced properties. In specific, we propose two clustering algorithms, called I-ESC and DI-ESC, which can deal with the subspace-imbalanced data sets as well as subspace-imbalanced and evolving data streams, respectively. Compared with ESC algorithm, I-ESC achieves more stable performance by using a more resonable method, *i.e.,* Eq. (7), to select the first exemplar instead of randomly selecting it like ESC. We prove that I-ESC is more capable of handing with subspace-imbalanced datasets than ESC. For subspace-imbalanced data streams, we further propose the DI-ESC algorithm which can perform subspace clustering in an online clustering manner. In addition, we also design a framework for DI-ESC to detect possible evolution in the data streams, making DI-ESC more capable of processing data streams with subspace emergence, disappearance, and recurrence. We verify the proposed algorithms on real-world data collected from different radar emitters.

Acknowledgment

The authors would like to thank the authors of [19], [20] and [18] for providing their source codes.

References

[1] W. Chen, K. Fu, J. Zuo, X. Zheng, T. Huang, and W. Ren, "Radar emitter classification for large data set based on weighted-xgboost," *IET Radar, Sonar Navigat.*, vol. 11, no. 8, pp. 1203–1207, Aug. 2017.

[2] M. Gupta, G. Hareesh, and A. K. Mahla, "Electronic warfare: Issues and challenges for emitter classification," *Defence Sci. J.*, vol. 61, no. 3, pp. 228–234, 2011.

[3] Z.-M. Liu, "Online pulse deinterleaving with finite automata," *IEEE Trans. Aerosp. Electron. Syst.*, early access, Jul. 27, 2019, doi: 10.1109/TAES.2019.2925447.

[4] Z.-M. Liu and P. S. Yu, "Classification, denoising, and deinterleaving of pulse streams with recurrent neural networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 4, pp. 1624–1639, Aug. 2019.

[5] G. Lopez-Risueno, J. Grajal, and A. Sanz-Osorio, "Digital channelized receiver based on time-frequency analysis for signal interception," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 879–898, Jul. 2005.

[6] C. Bertoncini, K. Rudd, B. Nousain, and M. Hinders, "Wavelet fingerprinting of radio-frequency identification (RFID) tags," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4843–4850, Dec. 2012.

[7] L. Li, H.-B. Ji, and L. Jiang, "Quadratic time–frequency analysis and sequential recognition for specific emitter identification," *IET Signal Process.*, vol. 5, no. 6, p. 568, 2011.

[8] J. Zhang, F. Wang, O. A. Dobre, and Z. Zhong, "Specific emitter identification via Hilbert–Huang transform in single-hop and relaying scenarios," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1192–1205, Jun. 2016.

[9] U. Satija, N. Trivedi, G. Biswal, and B. Ramkumar, "Specific emitter identification based on variational mode decomposition and spectral features in single hop and relaying scenarios," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 3, pp. 581–591, Mar. 2019.

[10] L. Ding, S. Wang, F. Wang, and W. Zhang, "Specific emitter identification via convolutional neural networks," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2591–2594, Dec. 2018.

[11] Y. Yuan, H. Wu, X. Wang, and Z. Huang, "Specific emitter identification based on Hilbert–Huang transform-based time–frequency–energy distribution features," *IET Commun.*, vol. 8, no. 13, pp. 2404–2412, Sep. 2014.

[12] Z. Yang, W. Qiu, H. Sun, and A. Nallanathan, "Robust radar emitter recognition based on the three-dimensional distribution feature and transfer learning," *Sensors*, vol. 16, no. 3, p. 289, 2016.

[13] J. Liu, J. P. Y. Lee, L. Li, Z.-Q. Luo, and K. M. Wong, "Online clustering algorithms for radar emitter classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1185–1196, Aug. 2005.
[14] J. Sun, G. Xu, W. Ren, and Z. Yan, "Radar emitter classification based on unidimensional convolutional neural network," *IET Radar, Sonar Navigat.*, vol. 12, no. 8, pp. 862–867, Aug. 2018.
[15] Y. Gong, G. Hu, and Z. Pan, "Structured sparsity preserving projections for radio transmitter recognition," in *Proc. Int. Conf. Mobile IT Converg.*, Sep. 2011, pp. 68–73.
[16] J. Ma, G. Huang, W. Zuo, X. Wu, and J. Gao, "Robust radar waveform recognition algorithm based on random projections and sparse classification," *IET Radar, Sonar Navigat.*, vol. 8, no. 4, pp. 290–296, 2013.
[17] C. You, C. Li, D. P. Robinson, and R. Vidal, "Scalable exemplar-based subspace clustering on class-imbalanced data," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 67–83.
[18] X. Peng, H. Tang, L. Zhang, Z. Yi, and S. Xiao, "A unified framework for representation-based subspace clustering of out-of-sample and large-scale data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2499–2512, Dec. 2016.
[19] R. Hyde, P. Angelov, and A. R. MacKenzie, "Fully online clustering of evolving data streams into arbitrarily shaped clusters," *Inf. Sci.*, vols. 382–383, pp. 96–114, Mar. 2017.
[20] X. Zhang, C. Furtlehner, C. Germain-Renaud, and M. Sebag, "Data stream clustering with affinity propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 7, pp. 1644–1656, Jul. 2014.
[21] K. I. Talbot, P. R. Duley, and M. H. Hyatt, "Specific emitter identification and verification," *Technol. Rev.*, vol. 113, pp. 113–133, Jan. 2003.
[22] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Progr. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Nov. 2016.
[23] C.-G. Li, C. You, and R. Vidal, "Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2988–3001, Jun. 2017.
[24] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.
[25] C. You, D. P. Robinson, and R. Vidal, "Provable self-representation based outlier detection in a union of subspaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3395–3404.
[26] M. C. V. Nascimento and A. C. P. L. F. de Carvalho, "Spectral methods for graph clustering—A survey," *Eur. J. Oper. Res.*, vol. 211, no. 2, pp. 221–231, Jun. 2011.
[27] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Ann. Data Sci.*, vol. 2, no. 2, pp. 165–193, Jun. 2015.
[28] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
[29] J. Sui *et al.*, "Sparse subspace clustering for evolving data streams," in *Proc. ICASSP-IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 7455–7459.

Jinping Sui was born in Jilin, China, in 1990. He received the B.S. degree in communication engineering from Northeastern University in 2013, and the M.S. degree in information and communication of engineering from the College of Electronic Science, National University of Defense Technology (NUDT), Changsha, China, in 2015, where he is currently pursuing the Ph.D. degree with the College of Electronic Science and Engineering. Since November 2017, he has been studying as a Visiting Ph.D. Student with the Machine Learning for Big Data research Group, Department of Computer Science, Aalto University, Finland. His research interests include data streaming and machine learning for big data.

Zhen Liu was born in Jiangsu, China, in 1983. He received the B.S. degree from Zhejiang University, Hangzhou, China, in 2006, and the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013. Since 2013, he has been a Lecturer with the College of Electronic Science and Engineering, NUDT. His research interests include signal processing, compressed sensing, and machine learning.

Li Liu received the B.Sc. degree in communication engineering, the M.Sc. degree in photogrammetry and remote sensing, and the Ph.D. degree in information and communication engineering from the National University of Defense Technology (NUDT), China, in 2003, 2005, and 2012, respectively. She joined the Faculty of NUDT in 2012, where she is currently an Associate Professor with the College of System Engineering. During her Ph.D. study, she spent more than two years as a Visiting Student with the University of Waterloo, Canada, from 2008 to 2010. From 2015 to 2016, she spent ten months visiting the Multimedia Laboratory, The Chinese University of Hong Kong. From 2016 to 2018, she worked as a Senior Researcher with the Machine Vision Group, University of Oulu, Finland. Her papers have currently over 2500 citations in Google Scholar. Her current research interests include computer vision, pattern recognition, and machine learning. She was the Co-Chair of nine International Workshops at CVPR, ICCV, and ECCV. She serves as the Area Chair of ACCV and ICME. She was a Guest Editor of special issues for the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and the *International Journal of Computer Vision*. She currently serves as an Associate Editor of *The Visual Computer* journal and *Pattern Recognition Letter*.

Bo Peng received the B.S., M.S., and Ph.D. degrees in information and communication engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2008, 2010, and 2014, respectively. He is currently a Lecturer with the College of Electronic Science and Engineering, NUDT. His current research interests include signal processing, micro-Doppler signature analysis, and pattern recognition.

Tianpeng Liu received the B.Eng., M.Eng., and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2008, 2011, and 2016, respectively. He is currently a Lecturer with the College of Electronic Science and Engineering, NUDT. His primary research interests are radar signal processing, electronic countermeasure, and cross-eye jamming.

Xiang Li was born in Hunan, China, in 1967. He received the B.S. degree from Xidian University, Xi'an, China, in 1989, and the Ph.D. degree from the National University of Defense Technology (NUDT) in 1998. He is currently a Professor with the NUDT. His research interests include signal processing, automation target recognition, and machine learning.