




## Article

# Multivariate Time Series Feature Extraction and Clustering Framework for Multi-Function Radar Work Mode Recognition

Ruozhou Fan <sup>1</sup>, Mengtao Zhu <sup>2,3</sup> and Xiongkui Zhang <sup>1,\*</sup>

<sup>1</sup> School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China; 3220225129@bit.edu.cn

<sup>2</sup> School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China; zhumentao@bit.edu.cn

<sup>3</sup> Laboratory of Electromagnetic Space Cognition and Intelligent Control, Beijing 100191, China

\* Correspondence: zhangxiongkui@bit.edu.cn

**Abstract:** Multi-Function Radars (MFRs) are sophisticated sensors with great agility and flexibility in adapting their transmitted waveform and control parameters. The recognition of MFR work modes based on the intercepted pulse sequences plays an important role in interpreting the functional purpose and threats of a non-cooperative MFRs. However, due to the increased flexibility of MFRs, radar work modes with emerging new modulations and control parameters always appear, and the supervised classification method suffers performance degradation or even failure. Unsupervised learning and clustering of MFR pulse sequences becomes urgent and important. This paper establishes a unified multivariate MFR time series feature extraction and clustering framework for MFR work mode recognition. At first, various features are collected to form the feature set. The feature set includes features extracted through deep learning based on recurrent auto-encoders, multidimensional time series toolkit features, and manually crafted features for radar inter-pulse modulations. Subsequently, several feature selection algorithms, combined with different clustering and classification methods, are used for the selection of an “optimal” feature subset. Finally, the effectiveness and superiority of the proposed framework and selected features are validated through simulated and measured datasets. In the simulated dataset containing 20 classes of work modes, under the most severe non-ideal conditions, we achieve a clustering purity of 73.46% and an NMI of 84.28%. In the measured dataset with seven classes of work modes, we achieve a clustering purity of 86.96% and an NMI of 90.10%.

**Keywords:** electronic warfare; working modes recognition; feature selection; non-dominated sorting genetic algorithm (NSGA-II); multivariate time series clustering



**Citation:** Fan, R.; Zhu, M.; Zhang, X. Multivariate Time Series Feature Extraction and Clustering Framework for Multi-Function Radar Work Mode Recognition. *Electronics* **2024**, *13*, 1412. <https://doi.org/10.3390/electronics13081412>

Academic Editors: Zenghui Zhang, Junfeng Wang and Hao Yan

Received: 17 February 2024

Revised: 31 March 2024

Accepted: 2 April 2024

Published: 9 April 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the development of electronic scanned array such as phased array, modern Multi-Function Radars (MFRs) are capable of performing multiple simultaneous tasks in the timeline. Moreover, MFRs can adaptively select or optimize inter- and intra-pulse modulations as well as control parameters in real time upon sensing their working environment and to meet higher-level mission demands [1–4]. The modulation patterns of each specific task feature have significant variability in control parameters such as Pulse Repetition Interval (PRI), Pulse Width (PW), and Radio Frequency (RF) [5–7]. Accurately recognizing and analyzing the work modes of MFR is difficult and has brought urgent challenges to modern electronic receivers.

The first and preliminary step for effectively recognizing the work modes and functional intentions of an MFR is to effectively model the system. Generally, MFRs are complex systems, and their signal generation mechanism can be modeled in a hierarchical way [6–11]. Syntactic models are the early attempts to describe the behaviors of MFRs [6,8,10]. However, first, hierarchical model-based recognition requires prior information of all the basic

elements and their transition rules in each hierarchical layer, which can hardly be fully obtained in non-cooperative applications. Second, these recognition methods codify all acquired prior knowledge into an almost fixed matching template and fit intercepted signals with the context of the priors subsequently, which can hardly follow the agility in the programmable parameters of newly developed MFRs, such as a radio-defined radar or a cognitive radar [12–15]. Then, the inter pulse modulation patterns of some control parameters (namely PRI, RF, and PW) are considered as more inherent characteristics to distinguish different work modes, which provides a promising direction to explore the work intentions of MFRs [16]. With recent advances of machine learning and deep learning, many inter-pulse modulation classification methods have been designed to deal with the recognition of modern MFRs [16–28]. In the author's previous study [29], a sequence-to-sequence classifier is proposed to solve the pulse-level recognition problem for work modes defined as different modulation combinations of multiple control parameters. The fine-grained recognition results can further reveal the mode transition of an MFR. From these studies, a conclusion may be drawn that the supervised classification of inter-pulse modulations can be solved through deep neural networks efficiently and accurately.

However, supervised learning with pre-acquired training data constrain their potential in a realistic application for MFR signals. First, acquiring sufficient training data for numerous and programmable work modes in real complex electromagnetic environments is complicated, labor-intensive and even impossible for some special modes hidden only for emergency scenarios. Second, a deep learning classifier pre-trained through supervised learning would never be guaranteed to be efficacious under the considerations that novel adaptive sequence patterns would constantly emerge along with the development of MFR itself. An increased degree of freedom of an MFR would pose more challenges for supervised recognition methods. Last but not least, cognitive MFRs are on their way to reality [12,14,15,30,31], which could work in a more fine-grained mode with the same modulation but different parameters to meet different performance requirements. It is of great importance to investigate recognition methods that require less prior information.

Generally, intercepted radar pulse sequences are represented using Pulse Descriptor Word (PDW) sequences. In terms of unsupervised learning, Guan [32] employs the concept of the Needleman-Wunsch algorithm and utilizes sequence alignment processing to obtain the MFR search mode rules. Fang [33] proposed an unsupervised change-point detection algorithm based on the Bayes criterion to recognize MFR work modes. Liu [34] proposed a semantic encoding model and encoding strategy optimization method for MFR pulse sequences. This method can automatically discover sequence patterns of MFR pulse sequence signals and represent them in the simplest form for extraction. However, these methods are applicable to a limited range of MFR work modes. Further research is needed to adapt them to new radar systems with modulation such as agile modulation. Taking the PDW sequences as Multivariate Time Series (MTS), unsupervised feature extraction and clustering methods can be investigated for a more general applied and less prior required solution. There are recent studies considering the time series clustering of radar signals [22,35]. Guillaume [35] focus on clustering pulse sequences from different radar emitters, and the mean value is used to represent the time series characteristics of a pulse sequence. Their method achieves satisfactory performance as the parameter values of different emitters are differentiable in high-dimensional PDW spaces. However, for an MFR, the parameter values of different work modes are close or even overlap; these methods would suffer performance degradation. In [22], parametric models for different PRI modulations are established, and three clustering methods are proposed for sub-sequence clustering of MFR work modes. However, clustering of multivariate time series requires further investigation. In fact, the clustering of multivariate time series has been investigated in many other fields. There are many comprehensive reviews for time series clustering [36–40] and a variety of investigations of multi-variate time series feature extraction or clustering methods [41–49]. However, to the best of our knowledge, there is still a lack of investigations into the recognition of MFR pulse sequences from a multivariate time series clustering perspective.

This paper puts its focus on establishing a multivariate time series feature extraction and clustering framework for MFR pulse sequences and selecting the optimal features for MFR work mode recognition. The clustering framework consists of five steps: pre-processing, feature extraction, feature selection, recognition, and evaluation. In the feature extraction part, manual feature and deep learning feature extraction methods are separately studied, including (1) feature engineering from hand-craft features for PRI modulation-type identification, (2) feature engineering from extensively designed MTS hand-craft features, and (3) unsupervised automatic feature engineering with deep neural networks. Extensive experiments are conducted to verify the effectiveness of the proposed clustering framework. Experimental results validate the superiority of the proposed framework and the effectiveness of the selected features. The main contributions of this paper can be summarized as follows:

- (1) A multivariate time series feature extraction and clustering framework is designed for MFR pulse sequences.
- (2) Several different implementations of the proposed framework are evaluated and compared. In each implementation, effective and advanced methods are utilized.
- (3) The experimental dataset includes a rich variety of radar modulation patterns; therefore, the selected features possess better universality.

The rest of this paper is organized as follows. Section 2 describes the formulation of the feature extraction and clustering tasks for multivariate MFR time series. Section 3 introduces the proposed framework and the corresponding implementations. Data description, experimental design, and experimental results are provided in Section 4. Finally, the conclusions and guidelines for future work are provided in Section 5.

## 2. Problem Formulation

This paper aims to implement a multivariate time series clustering framework for MFR pulse sequences. This section describes the definition of MFR work modes, provides definitions of work mode sequences, and presents mathematical formulation of the recognition task.

### 2.1. MFR Work Modes Definition

To serve multiple radar functions, each MFR work mode class has different intra- or inter-pulse parameters and can be defined as a certain arrangement of a finite or variable number of pulses. The author's previous study [29] defined the time series representation of an MFR work mode sequences which are briefly introduced here for completeness.

At first, an MFR work mode can be defined as modulation combinations on multiple control parameters [7,11,29,50], more specifically, in this study, the PRI, RF, and PW. Then, two layers of work modes can be derived to describe the ability of MFR and CR to adaptively select or optimize modulations or modulation parameters. The modulation-level work mode represents the fact that the MFR can select or optimize modulations or modulation parameters in corresponding modulation and parameter space. The final optimized result is denoted as parameter-level work mode and it can be seen as an implementation or instantiation of the modulation-level work mode. An input MTS with  $n$  pulses of the corresponding MFR work mode can be described as  $X \in \mathbb{R}^{M \times n}$ , where  $M$  is the number of parameters and each parameter in the pulse sequence obeys certain time series characteristics according to the modulation types.

There are different inter-pulse modulation styles for the three selected parameters. Six classes of modulation are employed in this study, including constant, agile, jittered, dwell and switch, sliding, and periodic. The candidate modulation types corresponding to the three parameters are listed in Table 1.

**Table 1.** Candidate modulation types of three control parameters PRI, RF, and PW.

Parameters	Candidate Modulation Types
PRI	constant, agile, jittered, dwell and switch, sliding, periodic
RF	constant, agile, jittered, dwell and switch, sliding
PW	constant, agile, jittered

2.2. Time Series Representation of an MFR Pulse Sequence

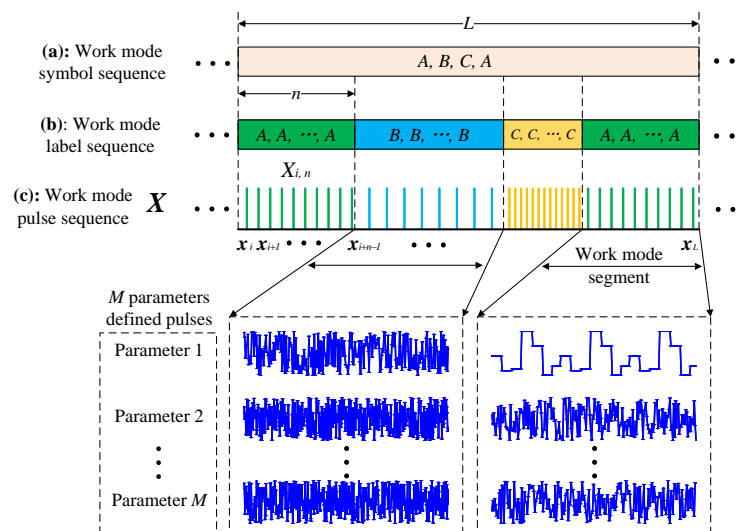
An MFR work mode sequence can be defined in a multiple-layer architecture, as described in [29]. Figure 1 illustrates the  $M$  parameters defining an MFR work mode sequence.

**Definition 1.** A radar pulse  $x \in \mathbb{R}^M$  is represented by a real-valued vector of  $M$  parameters as  $x = (x_1, x_2, \dots, x_M)^T$ .

**Definition 2.** A (work mode) pulse sequence  $X \in \mathbb{R}^{M \times L}$  of  $L$  pulses is a sequence of ordered pulses  $X = (x_1, x_2, \dots, x_L)$ .

**Definition 3.** A (work mode) segment  $X \in \mathbb{R}^{M \times n}$  of  $n$  pulses in  $X$  is a sub-sequence of pulses  $X_{i,n} = (x_i, x_{i+1}, \dots, x_{i+n-1})$ , where  $1 \leq i \leq L - n + 1, n \leq L$ . Each segment belongs to a certain class, while different segments may belong to the same class.

**Definition 4.** A (work mode) symbol sequence of  $J$  work mode segments and  $K$  work mode classes is a data structure that stores the work mode class symbols. A symbol represents the work mode class of a segment in  $X$ . For instance, an  $X$  with symbol sequence as ‘A, B, C, A’ contains four consecutive segments from three classes.



**Figure 1.** Illustration of  $M$  parameter-defined radar work mode sequence. From top to down (a) Work mode symbol sequence describing the segment level label for each segment in pulse sequence  $X$ , (b) Work mode label sequence describing the pulse level label for each pulse in  $X$ , (c) Work mode pulse sequence  $X$  containing all the pulses forming multiple work mode segments. Each segment in  $X$  is constituted with a sub-sequence of pulses, and each pulse is represented by an  $M$ -parameters vector.

Generally, the investigation of both feature extraction and clustering methods should utilize the radar segments from the individual work mode. Through such operation, the effectiveness of the extracted features and the performance of the clustering methods can be evaluated and compared with specific physical meanings. In applications, the input radar pulse sequence generally contains multiple segments from different work mode classes. In such a case, the investigated feature extraction and clustering methods can be

combined with sliding window techniques [16] for sub-sequence clustering [39] of pulse sequences with multiple work modes; this is another area of discussion. To keep this paper well focused, in the following sections, all methods are investigated in samples with a single work mode class.

### 2.3. Multivariate Time Series Clustering Task of MFR Pulse Sequences

An MFR pulse sequence clustering task is expected to output a cluster label for each input work mode sample. We let  $\mathcal{X} = \{X_1, X_2, \dots, X_N\}$  denote the MFR pulse sequence dataset with  $N$  pulse sequence samples. An input sample consisting of  $L$  pulses can be expressed as  $X_i = \{x_1, x_2, \dots, x_L\}$ ,  $1 \leq i \leq N$ , where  $x_t = \{x_1, x_2, \dots, x_M\}$ ,  $1 \leq t \leq L$  is the  $t$ th pulse, and  $M$  is the number of pulse parameters. The goal for the clustering task is to compute cluster labels  $\hat{Y} = \{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_N\}$  for  $\mathcal{X}$  based on the similarity of the samples.

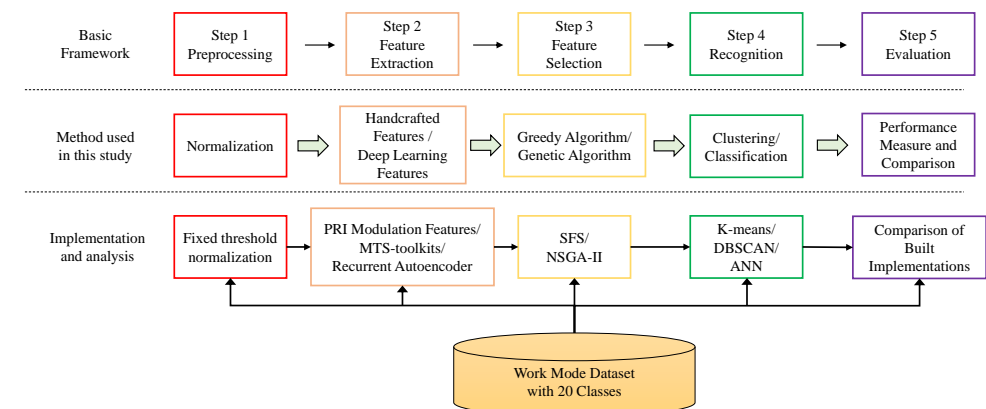
Generally, direct calculations of the similarity between  $(X_i, X_j)$ ,  $i \neq j$  are complicated due to great variability in number of pulses for different samples. Although dynamic time warping is a useful tool for similarity measuring of sequences with unequal length, its computational complexity is high for longer sequences. In this paper, the input pulse sequences with variable length are at first transformed to the feature space through function  $f_{feature} : \mathcal{X} \rightarrow \mathcal{H}$ , where  $\mathcal{H}$  is the extracted feature vector set. The clustering task is to divide feature dataset  $\mathcal{H}$  into  $K$  clusters by measuring the similarity in the feature space.

## 3. Methodology

Based on existing investigations on multivariate time series clustering, in this paper, a framework of feature extraction and unsupervised clustering is designed. This section at first describes the overall framework for MTS work mode clustering and then separately introduces the details of each step.

### 3.1. Framework Architecture

As shown in Figure 2, the general framework contains five steps including preprocessing, feature extraction, feature selection, recognition, and performance evaluation. In the feature extraction step, three sets of features are collectively extracted to form the entire feature set. Based on different method configurations in feature selection and recognition steps, different implementations are considered suitable for an MFR work mode pulse sequence.



**Figure 2.** The proposed multivariate time series feature extraction and clustering framework.

### 3.2. Preprocessing Method

Since a pulse is usually represented by multiple parameters (i.e.,  $M$  in this paper) which have different units and orders of magnitude, normalization is often necessary to bring all parameters to a comparable scale. A common normalization approach involves scaling parameter vector  $a$  based on its maximum and minimum values.

In the context of work mode recognition, a sequence may not encompass all work mode classes. Consequently, if normalization is solely based on the existing classes in

the sequence, the relative relationships between each class may be disturbed, leading to recognition errors. This impact becomes more pronounced during the testing phase, as received testing sequences are not guaranteed to contain all mode classes within a given time duration.

Hence, in this paper, pulse sequences are normalized based on fixed lower and upper bounds  $\mathbf{LB} = [LB_1, LB_2, \dots, LB_M]$  and  $\mathbf{UB} = [UB_1, UB_2, \dots, UB_M]$  for the  $M$  parameters, utilizing the following formula:

$$a'_m = \frac{2(a_m - LB_m)}{UB_m - LB_m} - 1, \quad m = 1, 2, \dots, M \quad (1)$$

where  $a_m$  represents the  $m$ th parameter vector of the input sequence and is normalized by the corresponding lower and upper bounds  $LB_m$  and  $UB_m$ . The values of  $\mathbf{LB}$  and  $\mathbf{UB}$  can adhere to the statistical range of pulse parameters. Therefore, normalizing with absent classes helps avoid issues of disrupting relationships.

### 3.3. Feature Extraction Methods

The sequence dataset with  $N$  samples  $X_i \in R^{M \times L_i}$ ,  $i = 1, 2, \dots, N$  is transformed to feature matrix  $\mathbf{H}_{initial} \in R^{n \times N}$ ;  $n$  is the number of extracted features.

#### 3.3.1. PRI Modulation Features

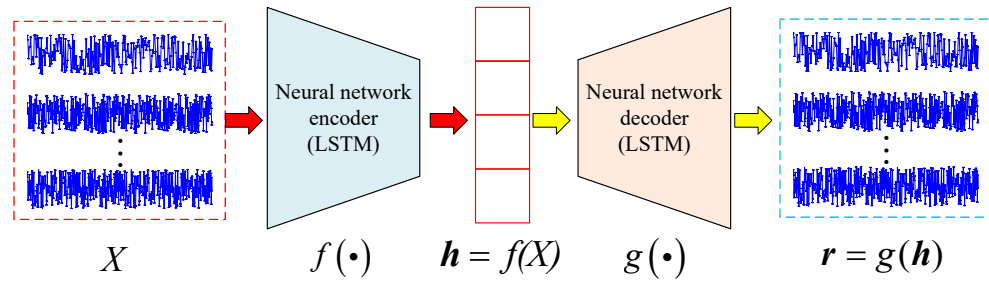
An efficient feature set for pulse repetition intervals modulation recognition (PRI Modulation Features) is proposed by researchers in [16]. The feature set includes several histograms and sequential features of PRI to describe specific modulation types. By cascading simple multilayer neural networks for classification, excellent modulation type recognition results can be achieved. In this study, PRI modulation features are extended to accommodate the condition of three parameters and the clustering task.

#### 3.3.2. MTS Features

We represent the radar work mode PDW sequence as an MTS. Therefore, we seek effective multivariate time series features for radar mode recognition. For example, in 2020, researchers in Georgia state university [51] presented a python toolkit for feature extraction of MTS which includes a comprehensive set of statistical features for extracting the important characteristics of MTS. In this paper, 39 kinds of these statistical features for each variate are extracted and added to the pool of candidate features.

#### 3.3.3. Unsupervised Neural Network Features

With the development of artificial intelligence and deep learning, automatic feature extraction become prevalent in pattern recognition. Autoencoders (AEs) are mature yet effective unsupervised feature extraction methods. For AEs, the encoder function at first encodes  $X$  to hidden representation  $\mathbf{h} = f(X)$ , then the decode function  $\mathbf{r} = g(\mathbf{h})$  decodes  $\mathbf{h}$  to reconstruct  $X$ . There are reconstruction errors between  $\mathbf{r}$  and  $X$ . The goal for an AE is to minimize the reconstruction error,  $\|\mathbf{r} - X\|_2$ . Generally,  $f(\bullet)$  and  $g(\bullet)$  are non-linear mappings, and thus AE is considered to extract more general and robust features. Since the raw MFR pulse sequences are with variable length and are multivariate time series, Recurrent AE (RAE) is utilized to extract time series features. The main difference between RAE and AE is that the encoder and decoder of RAE are Long Short-Term Memory (LSTM) layers, while for AE they are fully connected layers. LSTM layers are inherently suitable for extraction of time series features from raw time series data [52,53]. The structure of RAE in this paper is shown in Figure 3.



**Figure 3.** Recurrent autoencoder for unsupervised automatic feature extraction from raw MFR pulse sequence.

We let  $\mathcal{X} = \{X_1, X_2, \dots, X_N\}$  denote the MFR pulse sequence dataset with  $N$  pulse sequence samples. An input sample consisting of  $L$  pulses, which can be expressed as  $X_i = \{x_1, x_2, \dots, x_L\}, 1 \leq i \leq N$ . The goal for the RAE is to compute  $R = \{r_1, r_2, \dots, r_N\}$  for  $\mathcal{X}$  based on  $f(\bullet)$  and  $g(\bullet)$ . The RAE is trained to minimize reconstruction errors  $E_{RAE}$  in the training dataset. The loss function is expressed as follows:

$$E_{RAE} = \frac{1}{N} \sum_{i=1}^N \|g(f(X_i)) - X_i\|_2 \tag{2}$$

After the RAEs are trained, the output of encoder function  $h = f(X)$  when receiving testing pulse sequence  $X$  is treated as the extracted features.

### 3.4. Feature Selection Methods

The feature selection process is formulated as a multi-objective optimization (MOO) problem with selecting fewer MTS features and higher clustering performance as two optimizing objectives. The first objective function is  $f_1(t) = D(t)$ , where  $D(t)$  denotes the performance evaluation metrics of the clustering results for a given feature combination. The second objective function is the total number of the selected variables,  $f_2 = \sum_{i=1}^n t_i$ . Thus, the MOO problem can be formulated as follows:

$$\max_{t \in T} f(t), f(t) = (f_1(t), -f_2(t))^T \tag{3}$$

$$s. t. f_1(t) = D(t) \tag{4}$$

$$f_2(t) = \sum_{i=1}^n t_i \tag{5}$$

$$T = \{t \in \mathbb{R}^n | t_i(t_i - 1) = 0, i = 1, 2, \dots, n\} \tag{6}$$

Wherein decision vector  $t = (t_1, t_1, \dots, t_n)$  is used to formulate the identification of important features, where

$$t_i = \begin{cases} 1, & \text{ith feature selected} \\ 0, & \text{ith feature unselected} \end{cases} \quad i = 1, 2, \dots, n \tag{7}$$

The decision vector controls the selection of the subset of features for evaluation. Two methods are utilized and compared to solve the MOO problem for feature selection. The greedy search algorithm based on sequential forward selection and the heuristics method based on the Non-dominated Sorting Genetic Algorithm.

#### 3.4.1. Sequential Forward Selection (SFS)

Sequential forward selection is a greedy algorithm with fast solution speed and low time complexity. In the SFS, feature subset  $F$  starts from the empty set and adds one

current optimal feature  $f$  to  $F$  each time to make the feature selection  $J(F)$  the local optimal. The general process of SFS implementation is shown in Figure 4, Algorithm 1.

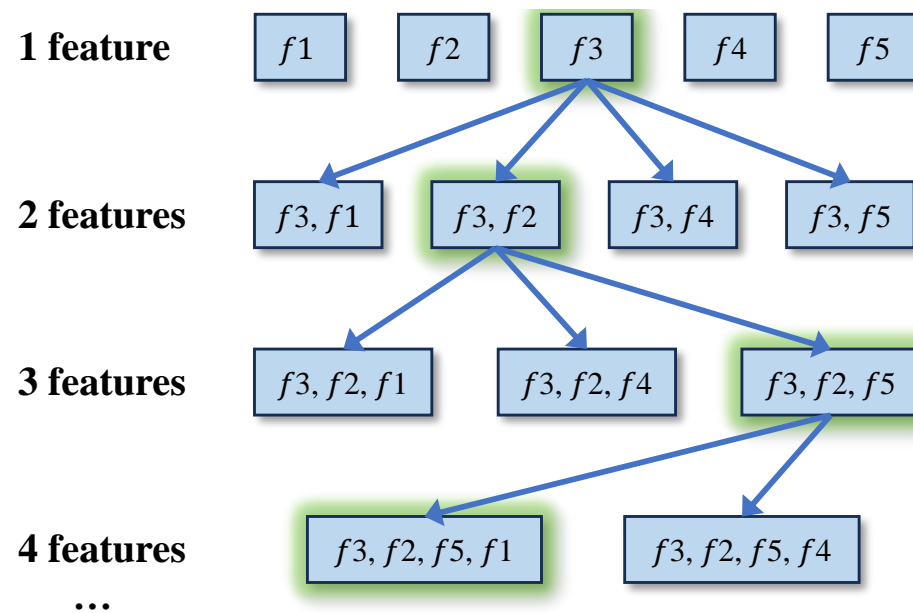


Figure 4. Sequential Forward Selection (SFS) process.

---

#### Algorithm 1 Sequential Forward Selection (SFS)

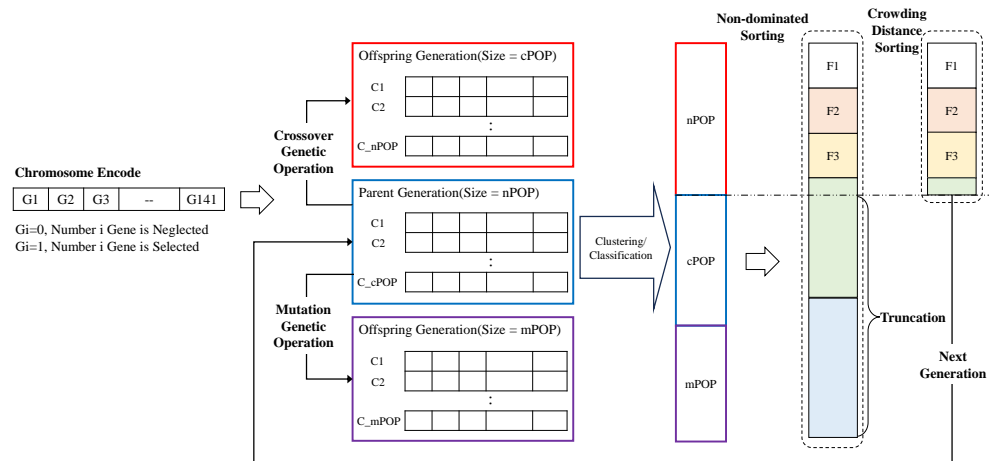
---

- 1: Input complete feature set  $Y$ , maximum number of feature subsets  $K$
  - 2: Initially the feature subset  $F$  to an empty set.
  - 3: In each iteration, select feature  $f$  from the complete feature set  $Y$  and add it to  $F$  such that the feature evaluation function  $J(F)$  achieves the maximum value.
  - 4: Check whether the current number of features  $k$  is equal to the desired number of feature subsets  $K$ .
  - 5: If yes, stop; otherwise, repeat the previous step until the condition is satisfied.
- 

#### 3.4.2. Non-Dominated Sorting Genetic Algorithm (NSGA-II)

A genetic algorithm is a heuristics search method for solving MO problems. The non-dominated sorting-based multi-objective evolutionary algorithm (NSGA-II) [54,55] is utilized in this study. NSGA-II adapts a suitable automatic mechanism based on the crowding distance (CD) to guarantee the diversity and spread of its solutions. The NSGA-II chromosome is encoded in a 141-bit binary sequence. Each bit represents a corresponding feature, and the whole string represents some combination of the candidate feature set. For population initialization, parents with nPOP chromosome sizes are randomly generated. In each subsequent cycle, through genetic operators of crossover and mutation, two offspring populations are generated from the corresponding parent. The sizes of the two offspring populations are cPOP and mPOP, respectively. For each individual in the simulated population, all "1" bit fields in the chromosome will be retrieved from the original feature set and connected to the clustering or classification input. Then based on the output of two objective functions, better qualified chromosomes are chosen through the NSGA-II algorithm. At the end of the simulation of the iterations, the algorithm converges to the best chromosomes that represent the optimal or sub-optimal solutions. The general process of NSGA-II implementation is shown in Figure 5, Algorithm 2.





**Figure 5.** Non-dominated Sorted Genetic Algorithm (NSGA-II) process.

---

**Algorithm 2** Non-dominated Sorting Genetic Algorithm (NSGA-II)

---

- 1: Input complete feature set  $Y$ , maximum number of iteration limit  $L$
  - 2: Initially, randomly select a subset of features from the original feature set as the first-generation population  $P$ .
  - 3: In each iteration, select excellent individuals by sequentially comparing the *Rank* labels and Crowding Distance (CD) values of two individuals.
  - 4: Apply crossover and mutation operations to generate a new generation population  $Q$ .
  - 5: Merge  $P$  and  $Q$  into a combined population  $R$ , and similarly use two levels of preference operators to select the optimal  $N$  individuals from  $R$  as the next-generation population.
  - 6: Check whether the preset iteration limit  $L$  is reached.
  - 7: If yes, stop; otherwise, repeat the previous step until the condition is satisfied
- 

### 3.5. Recognition Methods

Although our proposed framework primarily operates through unsupervised clustering methods for MFR work mode recognition, supervised classification methods are also included in the recognition process to validate the effectiveness of the selected features. There are already many mature yet effective clustering methods including distance-based, density-based, and spectral based ones [56]. In this study, one distance-based clustering method (K-means) and one density-based clustering algorithm (Density-Based Spatial Clustering of Applications with Noise, DBSCAN) are employed in the clustering step. K-means models are widely used in radar signal sorting [57] or clustering of different radar emitters [58]. DBSCAN is a density-based spatial clustering of applications with a noise algorithm [59] which does not require the priors of the number of clusters. During clustering, various distance metrics such as Euclidean distance and Cityblock distance are experimented with. In the end, Euclidean distance is selected for the experiments. Simultaneously, artificial neural networks are employed as a classification method to validate the effectiveness of the features.

## 4. Experiment and Analysis

In order to verify the effectiveness and superiority of the proposed framework and selected features, experiments with simulated MFR work mode pulse sequences were conducted. The experimental design, the datasets, and the evaluation metrics are described in Section 4.1. Then, the experimental results and the discussions are presented in Sections 4.2–4.4.

### 4.1. Experimental Design

#### 4.1.1. Dataset Description

According to Section 2.1, 20 classes of MFR work modes are considered (i.e.,  $K = 20$ ) based on different combinations of inter-pulse modulations on Pulse Repetition Interval (PRI), Radio Frequency (RF), and Pulse Width (PW), as depicted in Table 1. Table 2 shows the corresponding value ranges of modulation parameters for PRI, RF, and PW, respectively. Three kinds of non-ideal conditions including measuring noise, lost pulse, and spurious pulse are considered in the experiments. The three basic non-ideal settings are Measuring Noise Only (MNO), Lost Pulse Only (LPO), and Spurious Pulse Only (SPO). The MNO adds Gaussian distributed measuring noises to PRI, RF, and PW, respectively. The Gaussian noises have zero means and standard deviations in common units as  $\sigma = [\sigma_{PRI}, \sigma_{RF}, \sigma_{PW}]$ . There are seven levels of measuring noises according to variations of  $\sigma$  as  $\sigma_{PRI} = [0:0.05:0.3] \mu\text{s}$ ,  $\sigma_{RF} = [0:0.5:3] \text{MHz}$ ,  $\sigma_{PW} = [0:0.05:0.3] \mu\text{s}$ . For instance, the first level is  $\sigma_1 = [0 \mu\text{s}, 0 \text{MHz}, 0 \mu\text{s}]$ , and the second level is  $\sigma_2 = [0.05 \mu\text{s}, 0.5 \text{MHz}, 0.05 \mu\text{s}]$ . Both LPO and SPO separately consider the pulse sequences with a proportion of lost or spurious pulses. There are seven levels for both lost pulse and spurious pulse proportions with a range of [0:5:30]%. In addition, seven hybrid scenes are defined to evaluate the joint influence of combined non-ideal situations as depicted in Table 3. Thus, there are seven datasets for MNO, LPO, SPL, and hybrid scenarios, respectively. For each work mode class in each dataset, 500 samples are simulated, and there are a total of 10,000 sequence samples for each dataset. The number of pulses in a work mode sample is set to 200. In addition to simulated data, we also collected some actual measured signals to form the measured scenarios.

**Table 2.** Parameter settings for the 20 work modes considered in this study.  $U(\bullet)$  denotes uniform distribution.

Variables	PRI	RF	PW
Pulse parameters			
Initial value interval	$U(100, 200) \mu\text{s}$	$U(9 \times 10^3, 9.2 \times 10^3) \text{MHz}$	$U(1, 50) \mu\text{s}$
Agile			
Number of bursts	$U(4, 8)$	$U(4, 8)$	$U(4, 8)$
Dwell and switch			
Number of bursts	$U(2, 8)$	$U(2, 8)$	–
Jittered			
Deviation	$U(5\%, 15\%)$	$U(5\%, 15\%)$	$U(5\%, 15\%)$
Sliding			
Number of bursts	$U(4, 8)$	$U(4, 8)$	–
Size of step	$U(5, 50) \mu\text{s}$	$U(-50, 50) \text{MHz}$	–
Periodic (Sinusoidal) Carrier frequency	$U(5, 200) \text{Hz}$	–	–
Amplitude	$U(2, 5)$	–	–
Deviation	$U(2, 4)$	–	–

**Table 3.** Seven scenarios with hybrid non-ideal situations.

Scene	Measuring Noise ( $\mu\text{s}$ , MHz, $\mu\text{s}$ )	Lost Pulse (%)	Spurious Pulse (%)
1	[0, 0, 0]	0	0
2	[0.05, 0.5, 0.05]	5	5
3	[0.1, 1, 0.1]	10	10
4	[0.15, 1.5, 0.15]	15	15
5	[0.2, 2, 0.2]	20	20
6	[0.25, 2.5, 0.25]	25	25
7	[0.3, 3, 0.3]	30	30

#### 4.1.2. Evaluation Metrics

Clustering purity and Normalized mutual information (NMI) are two fundamental evaluation metrics for assessing clustering performance, while cross-entropy loss serves as a fundamental metric for evaluating multi-class classification performance. This study employs these three metrics for performance evaluation. The descriptions of these three metrics are as follows:

(1) Purity

The clustering purity is defined as

$$\text{Purity}(\Omega, C) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j| \quad (8)$$

where  $N$  is the total number of samples.  $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$  denotes the clustering results.  $C = \{c_1, c_2, \dots, c_J\}$  denotes the real assignments.

(2) Normalized mutual information

Normalized mutual information can be described as

$$\text{NMI}(\Omega, C) = \frac{2I(\Omega; C)}{H(\Omega) + H(C)} \quad (9)$$

where  $I(\bullet)$  is mutual information and  $H(\bullet)$  is entropy. They are defined as [60]

$$I(\Omega; C) = \sum_k \sum_j P(\omega_k \cap c_j) \log \frac{P(\omega_k \cap c_j)}{P(\omega_k)P(c_j)} = \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|} \quad (10)$$

$$H(\Omega) = -\sum_k P(\omega_k) \log P(\omega_k) = -\sum_k \frac{|\omega_k|}{N} \log \frac{|\omega_k|}{N} \quad (11)$$

where  $P(\omega_k)$ ,  $P(c_j)$ , and  $P(\omega_k \cap c_j)$  denote the possibility of a sample belonging to cluster  $\omega_k$ , category  $c_j$ , and both of them, respectively.  $I(\Omega; C)$  represents the increase in cluster information  $\omega$  for given class information  $C$ . That is,

$$I(\Omega; C) = H(\Omega) - H(\Omega|C) \quad (12)$$

(3) Cross-entropy loss

Cross-entropy loss is a commonly used loss function in machine learning and particularly in the context of classification problems. It measures the performance of a classification model whose output is a probability value between 0 and 1. The goal of the model is to assign the correct label to each input. For a multi-class classification problem with  $K$  classes, the formula is a generalization:

$$L(y, \hat{y}) = -\sum_{i=1}^K y_i \bullet \log(\hat{y}_i) \quad (13)$$

where  $y_i$  is the indicator function that equals 1 if the true class is  $i$  and 0 otherwise,  $\hat{y}_i$  is the predicted probability that the instance belongs to class  $i$ .

#### 4.1.3. Experimental Design

Combining two different feature selection methods with three different recognition methods results in a total of six implementations. Therefore, experiments were conducted using each of these implementations individually as follows (Figure 6):

- (1) Feature selection results analysis for different optimization methods (Section 4.2).
- (2) Robustness against typical non-ideal situations (Section 4.3).
- (3) Performance against different numbers of MFR work mode classes (Section 4.4).

(4) Performance validation with measured signals (Section 4.5).

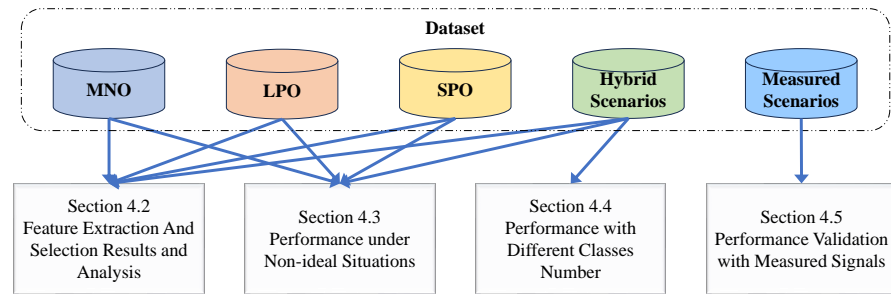


Figure 6. Experimental design and analysis.

#### 4.2. Feature Extraction and Selection Results and Analysis

In order to make the selected features more universally applicable, during the feature selection process, the dataset is formed using data from all 20 classes of working modes in all non-ideal condition scenario datasets. For each class of data in each scenario dataset, 50 random samples are selected and added to the complete dataset  $\mathcal{X}$ . Feature extraction is then performed on the dataset using the three feature extraction methods described in Section 3.3, resulting in a complete feature set  $Y$ . Figure 7 presents the t-Distributed Stochastic Neighbor Embedding (t-SNE) visualization results of all feature sets. From this, it can be observed that, before feature selection, different work mode classes are challenging to distinguish, indicating the presence of redundant features within the feature set.

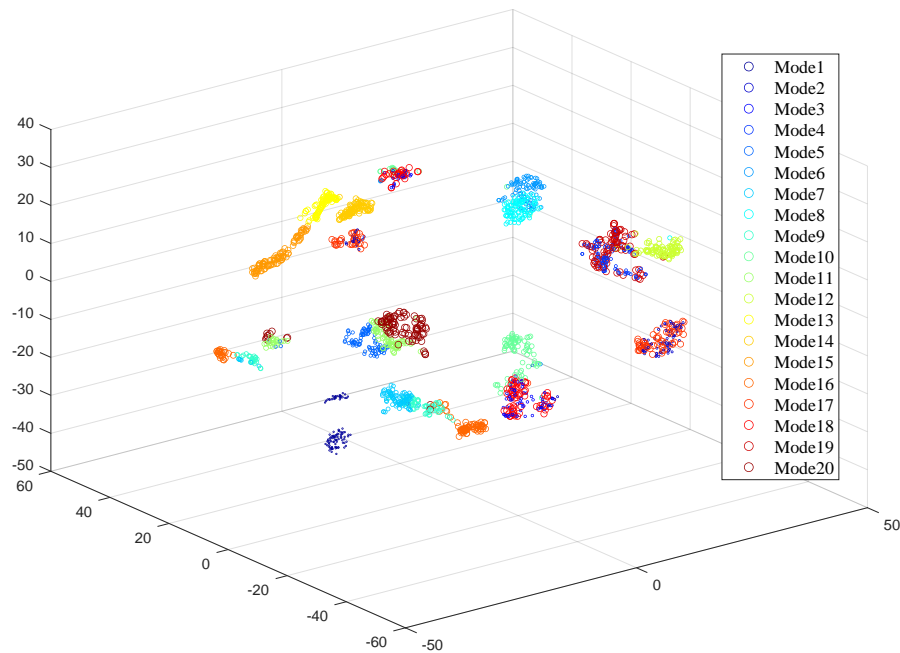


Figure 7. t-SNE visualization results of complete feature set.

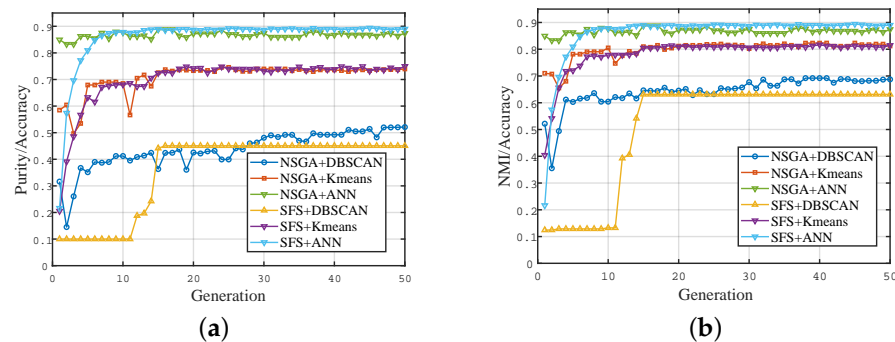
Based on six implementations composed of different feature selection and clustering methods, features of the MFR work mode are selected, and the results are depicted in the figure. The optimization objectives and maximum iteration count for each implementation are shown in the table below (Table 4).

Figure 8 illustrates the performance changes during the iterative process of feature selection for different implementations. It can be observed that the iteration performance of the NSGA-II algorithm is superior to that of the SFS algorithm. This is because NSGA-II possesses strong global search capabilities, allowing for it to find widely distributed solutions in the search space. In contrast, SFS is prone to becoming stuck in local optima,

especially when the clustering method is DBSCAN, quickly converging to a local optimum without further optimization. Additionally, since the K-means clustering method has a known number of clusters and strong prior information, its performance is better than that of the DBSCAN clustering method. As a supervised classification method, ANN exhibits the best performance and quickly reaches convergence. This demonstrates that the selected features have clear discriminability in a high-dimensional space.

**Table 4.** The algorithm parameter settings for the six implementations.

Implementation	Optimization Objectives $\max_{t \in T} f(t)$	Maximum Iteration/Maximum Number of Features
NSGA + DBSCAN	$f(t) = (f_1(t), -f_2(t))^T$ $f_1(t) = Purity(\Omega_t, C_t) + NMI(\Omega_t, C_t)$ $f_2(t) = \sum_{i=1}^n t_i$	50
NSGA + Kmeans	$f(t) = (f_1(t), -f_2(t))^T$ $f_1(t) = Purity(\Omega_t, C_t) + NMI(\Omega_t, C_t)$ $f_2(t) = \sum_{i=1}^n t_i$	50
NSGA + ANN	$f(t) = (f_1(t), -f_2(t))^T$ $f_1(t) = L(y_t, \hat{y}_t)$ $f_2(t) = \sum_{i=1}^n t_i$	50
SFS + DBSCAN	$f(t) = Purity(\Omega_t, C_t) + NMI(\Omega_t, C_t)$	50
SFS + Kmeans	$f(t) = Purity(\Omega_t, C_t) + NMI(\Omega_t, C_t)$	50
SFS + ANN	$f(t) = -L(y_t, \hat{y}_t)$	50



**Figure 8.** The iterative process of the feature selection algorithm. (a) Purity/Accuracy, (b) NMI/Accuracy.

Table 5 presents the distribution of the selected number of features for each implementation. It can be observed that the proportion of PRI modulation features is relatively high, indicating that features designed specifically for radar parameter inter-pulse modulation types are more effective. The relatively small proportion of MTS features is due to the fact that some features in this MTS feature extraction toolkit are only suitable for continuous time series, while PDW sequences are discrete and not suitable for these features. Finally, there is considerable room for improvement in features based on deep learning. In the future, network structures can be improved and regularization terms can be added to extract more effective features.

**Table 5.** The distribution of feature selection results.

Implementation	PRI Modulation Features Number	MTS Features Number	Unsupervised Neural Network Features Number
Complete Feature Set	15	106	20
NSGA + DBSCAN	10	3	2
NSGA + Kmeans	12	9	1
NSGA + ANN	9	4	1
SFS + DBSCAN	6	4	2
SFS + Kmeans	7	22	2
SFS + ANN	8	17	3

#### 4.3. Performance under Non-Ideal Situations

The pulse stream is often contaminated by highly non-ideal electromagnetic environments. A good method should be robust enough to correctly identify corrupted pulse sequences. This section evaluates the performance of different methods under non-ideal conditions, including three distinct non-ideal scenarios: MNO, LPO, SPL, and hybrid scenarios.

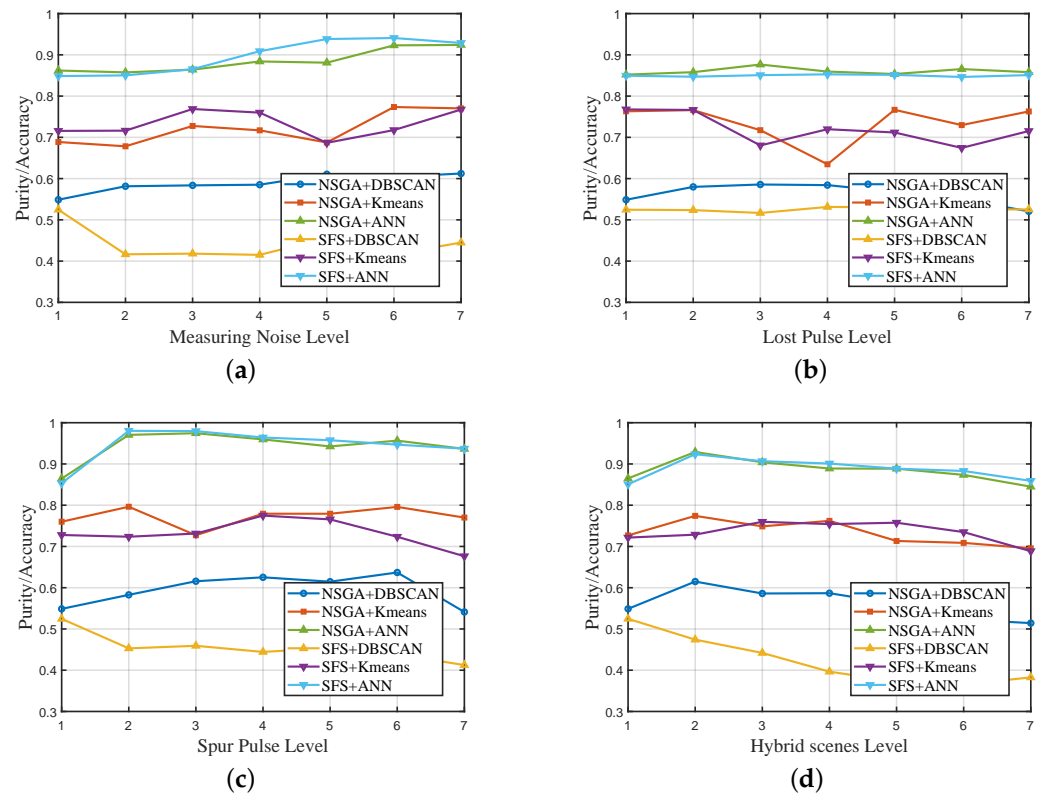
The clustering purity/classification accuracy for each implementation in different scenarios is displayed in Figure 9, separately. Due to the supervised nature of the ANN method, which involves utilizing more prior information, the classification accuracy is significantly higher than the clustering purity of other clustering methods. Different feature selection methods have little influence on the ANN method, and the classification accuracy remains above 84% in all scenarios.

In the MNO scenario, noise does not cause substantial negative effects on all implementations. In fact, there is even some performance improvement in measuring more severe noise conditions. The reason may be that when the noise is relatively low, the impact of different parameters of the same working mode is too significant, leading to a relatively large intra-class distance. As the noise increases, the intra-class distance becomes relatively smaller, making it easier to distinguish between different classes.

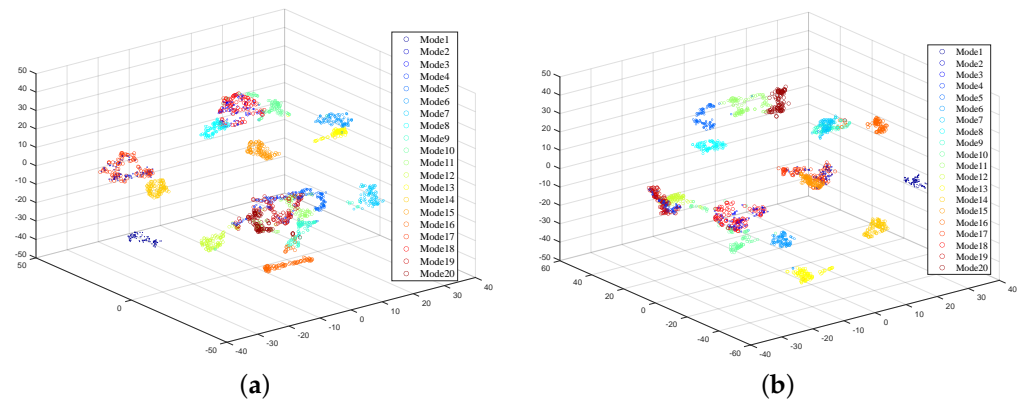
In the LPO scenario, each implementation can maintain stable performance, with relatively minor effects from changes in the proportion of missing pulses. The K-means clustering method, due to its sensitivity to the initialization of clustering, exhibits slightly larger performance fluctuations. The selected features and clustering implementations demonstrate good robustness to the situation of missing pulses.

For the SPL scenario and the hybrid scenario, the diversity introduced by spurious pulses increases the variability of sample features. Therefore, under non-ideal conditions, there are instances where performance may show some improvement. However, the overall performance in the mixed scenario tends to decrease as non-ideal conditions worsen.

In summary, the proposed implementations and the selected features exhibit good robustness in non-ideal scenarios, providing satisfactory distinctiveness in both unsupervised clustering and supervised classification. Among clustering implementations, NSGA+Kmeans performs better than others. Figure 10 shows the t-SNE visualization results of features extracted by implementations of NSGA+Kmeans and NSGA+ANN. It can be observed from the figure that features from the majority of work modes exhibit clear distinguishability, while there is some overlap in features from a few minority modes.



**Figure 9.** Performances of different implementations in different non-ideal situations. (a) Measuring noise only, (b) lost pulse only, (c) spurious pulse only, (d) hybrid scenes.



**Figure 10.** t-SNE visualization results of features extracted by implementations. (a) NSGA + Kmeans; (b) NSGA + ANN.

#### 4.4. Performance with Different Class Numbers

When comparing the effects of different implementations under various MFR work mode classes, Hybrid Scenario 4 was selected as the dataset. Two to twenty work mode classes were randomly selected from the dataset for each experiment.

Due to the different adaptability of different features to various classes, there is some fluctuation in the decreasing trend. Figure 11 illustrates the feature visualization when randomly selecting 4 working mode classes multiple times. It can be observed from the figure that the distinctiveness of features varies when different classes are randomly sampled. The selected features showed poor adaptability to mode2, mode8, mode19 and mode17. Therefore, 100 tests were conducted randomly for each work mode class number, and the averages were taken as the final results. The experimental results are shown in

Figure 12. It can be observed from the figure that as the number of work mode classes increases, the overall performance of each implementation shows a decreasing trend.

Similarly, the NSGA+Kmeans implementation exhibits the best performance among clustering implementations, achieving purity of over 73.46% and NMI of over 84.28%. On the other hand, the SFS and DBSCAN implementation performs the worst, with a minimum purity of 39.65% and an NMI of 50.93%.

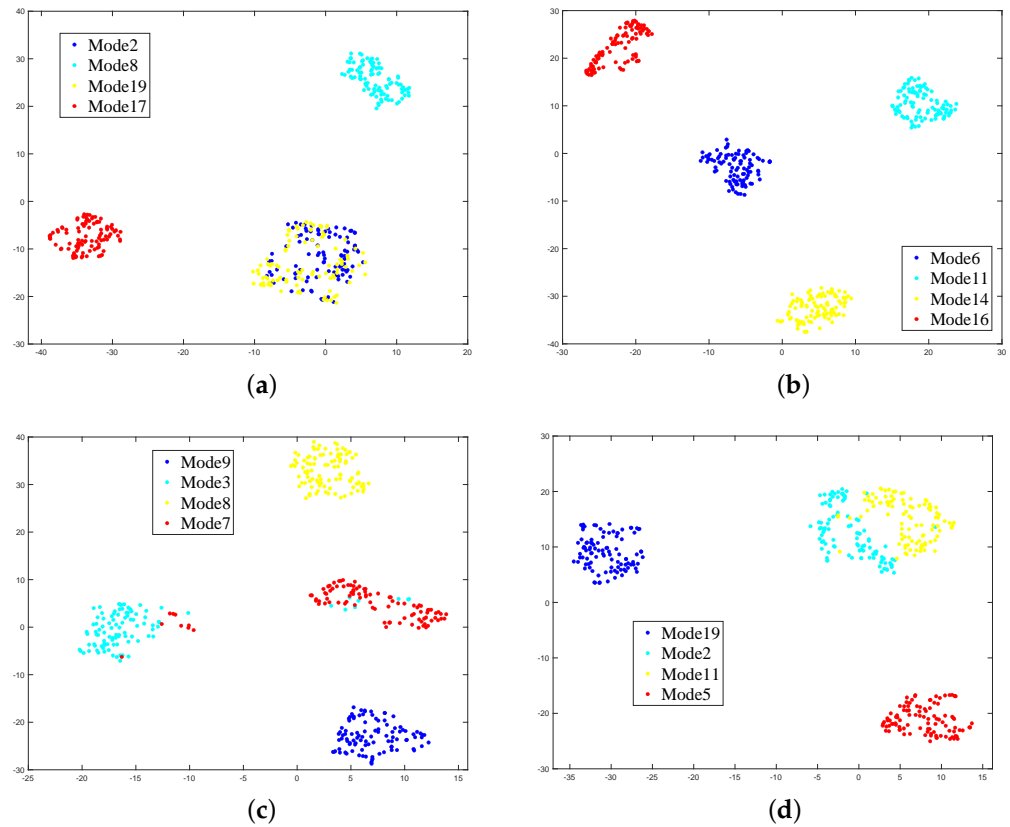


Figure 11. t-SNE visualization results of features extracted by implementation of NSGA+Kmeans. (a) Mode2, 8, 19, 17; (b) Mode6, 11, 14, 16; (c) Mode9, 3, 8, 7; (d) Mode19, 2, 11, 5.

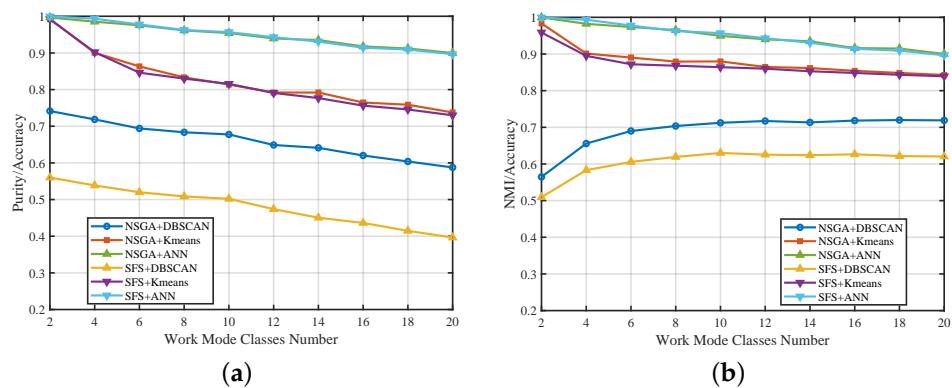


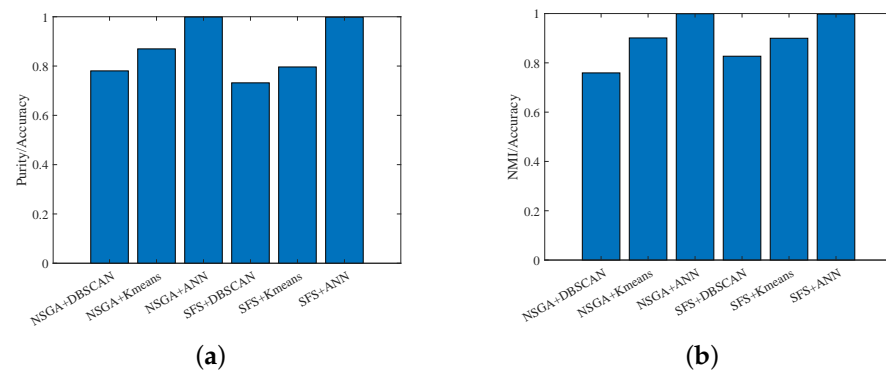
Figure 12. Performances of different implementations with different work mode class numbers. (a) Purity/Accuracy; (b) NMI/Accuracy.

#### 4.5. Performance Validation with Measured Signals

In fact, it is not convincing to evaluate the work mode recognition capability of these implementations only by testing them on simulated datasets. We used a radar simulator to generate signals and transmit them into space through a horn antenna, measuring 7 classes



of the radar work mode to form the measured dataset. The PRI ranged from 378 to 1728  $\mu\text{s}$ , the PW ranged from 16 to 93  $\mu\text{s}$ , and the RF ranged from 2750 to 9780 MHz. The elevation angle of the receiving antenna was approximately 10 degrees. The feature set selected in Section 4.2 was used for the work mode recognition of the measured signals. Figure 13 presents the recognition performance of different implementations. Both NSGA + ANN and SFS + ANN demonstrated satisfactory supervised recognition performance, achieving an almost 100% accuracy. As for the clustering methods, NSGA + Kmeans remained the best-performing approach, with clustering purity and NMI of 86.96% and 90.10%, respectively. The relatively poorest-performing approach, SFS + DBSCAN, had clustering purity and NMI of 73.19% and 82.69%, respectively. The proposed feature extraction and clustering framework and the selected feature set exhibited good work mode recognition capability.



**Figure 13.** Performances of different implementations with 7 measured work mode classes. (a) Purity/Accuracy; (b) NMI/Accuracy.

## 5. Conclusions

With development in optimization theory, computation ability, and software-defined system architecture, MFR work modes with unseen modulations and modulation parameters emerge persistently. Unsupervised clustering of an MFR pulse sequence become urgent and important for electronic reconnaissance systems.

In this paper, a unified clustering framework is established for multivariate MFR time series and feature selection is conducted on a large number of time series features. Based on the existing advancements in time series research and the machine learning community, in this paper, features extracted through autoencoder-based deep learning, multidimensional time series features, and manually crafted PRI-type recognition features are considered and utilized in the traditional domain to form a feature set. Following that, NSGA-II and SFS feature selection algorithms are applied in conjunction with various clustering and classification methods to optimize features. Ultimately, the proposed framework and the chosen features are validated for effectiveness and superiority through extensive simulations on pulse sequence datasets.

Unsupervised recognition of MFR work modes plays a significant role in modern electromagnetic environments due to the increased degree of freedom of modern MFRs. There are many future works that can be investigated. First, more adaptive and accurate clustering methods are required to conclude the irregular scattering of different work modes in a high-dimensional feature space. Second sub-sequence clustering methods should be investigated with the findings in this study for clustering of pulse sequences with multiple consecutive radar work modes. Finally, probabilistic graphical models need to be investigated for the possible dependence between different variables in MFR applications.

**Author Contributions:** Conceptualization, M.Z. and X.Z.; Methodology, R.F.; Software, R.F.; Validation, R.F.; Data curation, R.F.; Writing—original draft, R.F. and M.Z.; Writing—review & editing, M.Z. and X.Z.; Project administration, X.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation (NSFC) of China under Grant no. 62301031, Young Elite Scientists Sponsorship Program by BAST under Grant BYESS2023415.

**Data Availability Statement:** The original data presented in the study are openly available in ScienceDB at 10.57760/sciencedb.17706.

**Acknowledgments:** The authors thank the editors and anonymous referees for their efforts and constructive comments to improve the quality of this paper.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Weber, M.E.; Cho, J.Y.N.; Thomas, H.G. Command and Control for Multifunction Phased Array Radar. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5899–5912. [[CrossRef](#)]
2. Stailey, J.E.; Hondl, K.D. Multifunction Phased Array Radar for Aircraft and Weather Surveillance. *Proc. IEEE* **2016**, *104*, 649–659. [[CrossRef](#)]
3. Mir, H.S.; Guitouni, A. Variable Dwell Time Task Scheduling for Multifunction Radar. *IEEE Trans. Autom. Sci. Eng.* **2014**, *11*, 463–472. [[CrossRef](#)]
4. Boers, Y.; Driessen, H.; Zwaga, J. Adaptive MFR parameter control: Fixed against variable probabilities of detection. *IEE Proc. Radar Sonar Navig.* **2006**, *153*, 2–6. [[CrossRef](#)]
5. Arasaratnam, I.; Haykin, S.; Kirubarajan, T.; Dilkes, F.A. Tracking the mode of operation of multi-function radars. In Proceedings of the IEEE Conference on Radar, Verona, NY, USA, 24–27 April 2006.
6. Visnevski, N.; Krishnamurthy, V.; Wang, A.; Haykin, S. Syntactic Modeling and Signal Processing of Multifunction Radars: A Stochastic Context-Free Grammar Approach. *Proc. IEEE* **2007**, *95*, 1000–1025. [[CrossRef](#)]
7. Apfeld, S.; Charlish, A.; Ascheid, G. Modelling, Learning and Prediction of Complex Radar Emitter Behaviour. In Proceedings of the 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), Boca Raton, FL, USA, 16–19 December 2019; pp. 305–310. [[CrossRef](#)]
8. Visnevski, N.A. Syntactic Modeling of Multi-Function Radars. Ph.D. Thesis, McMaster University, Hamilton, ON, Canada, 2005.
9. Wang, A.; Krishnamurthy, V. Threat Estimation of Multifunction Radars: Modeling and Statistical Signal Processing of Stochastic Context Free Grammars. In Proceedings of the IEEE International Conference on Acoustics, Honolulu, HI, USA, 15–20 April 2007.
10. Wang, A.; Krishnamurthy, V. Signal Interpretation of Multifunction Radars: Modeling and Statistical Signal Processing With Stochastic Context Free Grammar. *IEEE Trans. Signal Process.* **2008**, *56*, 1106–1119. [[CrossRef](#)]
11. Ou, J.; Chen, Y.; Zhao, F.; Ai, X.; Yang, J. Research on extension of hierarchical structure for multi-function radar signals. In Proceedings of the 2017 Progress In Electromagnetics Research Symposium—Spring (PIERS), St. Petersburg, Russia, 22–25 May 2017; pp. 2612–2616. [[CrossRef](#)]
12. Haykin, S.; Xue, Y.; Setoodeh, P. Cognitive Radar: Step toward Bridging the Gap between Neuroscience and Engineering. *Proc. IEEE* **2012**, *100*, 3102–3130. [[CrossRef](#)]
13. Blunt, S.D.; Mokole, E.L. Overview of radar waveform diversity. *IEEE Aerosp. Electron. Syst. Mag.* **2016**, *31*, 2–42. [[CrossRef](#)]
14. Greco, M.S.; Gini, F.; Stinco, P.; Bell, K. Cognitive Radars: On the Road to Reality: Progress thus Far and Possibilities for the Future. *IEEE Signal Process. Mag.* **2018**, *35*, 112–125. [[CrossRef](#)]
15. Gurbuz, S.Z.; Griffiths, H.D.; Charlish, A.; Rangaswamy, M.; Greco, M.; Bell, K.L. An Overview of Cognitive Radar: Past, Present, and Future. *IEEE Aerosp. Electron. Syst. Mag.* **2019**, *34*, 6–18. [[CrossRef](#)]
16. Kauppi, J.P.; Martikainen, K.; Ruotsalainen, U. Hierarchical classification of dynamically varying radar pulse repetition interval modulation patterns. *Neural Netw.* **2010**, *23*, 1226–1237. [[CrossRef](#)] [[PubMed](#)]
17. Liu, Y.; Zhang, Q. An improved algorithm for PRI modulation recognition. In Proceedings of the 2017 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xiamen, China, 22–25 October 2017; pp. 1–5. [[CrossRef](#)]
18. Hu, G.; Liu, Y. An Efficient Method of Pulse Repetition Interval Modulation Recognition. In Proceedings of the International Conference on Communications & Mobile Computing, Shenzhen, China, 12–14 April 2010.
19. Mahmoud, K.; Amir Mansour, P.; Forouhar, F. A new method for detection of complex Pulse Repetition Interval Modulations. In Proceedings of the IEEE International Conference on Signal Processing, Beijing, China, 21–25 October 2012.
20. Li, X.; Huang, Z.; Wang, F.; Xiang, W.; Liu, T. Towards Convolutional Neural Networks on Pulse Repetition Interval Modulation Recognition. *IEEE Commun. Lett.* **2018**, *22*, 2286–2289. [[CrossRef](#)]
21. Liu, Z.M.; Yu, P.S. Classification, Denoising and Deinterleaving of Pulse Streams with Recurrent Neural Networks. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *55*, 1624–1639. [[CrossRef](#)]
22. Zhu, M.; Li, Y.; Wang, S. Model-based Time Series Clustering and Inter-pulse Modulation Parameter Estimation of Multi-function Radar Pulse Sequences. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 3673–3690. [[CrossRef](#)]
23. Wang, B.; Wang, S.; Zeng, D.; Wang, M. Convolutional Neural Network-Based Radar Antenna Scanning Period Recognition. *Electronics* **2022**, *11*, 1383. [[CrossRef](#)]

24. Li, W.; Dong, Y.Y.; Zhang, L.; Dong, C. Time Domain Attention Mechanism Based Multi-Functional Radar Working Mode Recognition. In Proceedings of the 2023 8th International Conference on Communication, Image and Signal Processing (CCISP), Chengdu, China, 17–19 November 2023; pp. 457–462. [CrossRef]
25. Wang, W.; Wang, Y. A Parameter-Optimized SLS-TSVM Method for Working Modes Recognition of Airborne Fire Control Radar. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; Volume 1, pp. 948–953. [CrossRef]
26. Tang, Z.; Gong, Y.; Tao, M.; Su, J.; Fan, Y.; Li, T. Recognition of Working Mode for Multifunctional Phased Array Radar Under Small Sample Condition. In Proceedings of the 2023 IEEE 6th International Conference on Electronic Information and Communication Technology (ICEICT), Qingdao, China, 7–9 August 2023; pp. 1157–1160. [CrossRef]
27. Gao, J.; Wang, T.; Ye, F. Radar Working Mode Recognition Algorithm Based on Siamese Network and Deep Auto Encoder-Affinity Propagation. In Proceedings of the 2022 International Symposium on Antennas and Propagation (ISAP), Sydney, Australia, 31 October–3 November 2022; pp. 371–372. [CrossRef]
28. Zhang, Z.; Shi, X.; Zhou, F. An Incremental Recognition Method for MFR Working Modes Based on Deep Feature Extension in Dynamic Observation Scenarios. *IEEE Sens. J.* **2023**, *23*, 21574–21587. [CrossRef]
29. Li, Y.; Zhu, M.; Ma, Y.; Yang, J. Work Modes Recognition and Boundary Identification of MFR Pulse Sequences with a Hierarchical Seq2seq LSTM. *Electronic Article*. 2020. Available online: <https://digital-library.theiet.org/content/journals/10.1049/iet-rsn.2020.0060> (accessed on 24 July 2020).
30. Charlish, A.; Hoffmann, F.; Degen, C.; Schlangen, I. The Development from Adaptive to Cognitive Radar Resource Management. *IEEE Aerosp. Electron. Syst. Mag.* **2020**, *35*, 8–19. [CrossRef]
31. Martone, A.F.; Charlish, A. Cognitive radar for waveform diversity utilization. In Proceedings of the IEEE Radar Conference, Atlanta, GA, USA, 7–14 May 2021.
32. Guan, X.; Zhang, Y.; Ling, H. Main pattern extraction of electronic scanning radar. *Electron. Opt. Control.* **2018**, *25*, 84–88. (In Chinese)
33. Fang, Y.; Bi, D.; Pan, J.; Chen, Q. Multi-function Radar Behavior State Detection Algorithm based on Bayesian Criterion. In Proceedings of the 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chengdu, China, 20–22 December 2019; IEEE: Piscataway, NJ, USA, 2020; pp. 213–217.
34. Liu, Z.; Yuan, S.; Kang, S. Semantic Coding and Model Reconstruction of Multifunction Radar Pulse Train. *J. Radars* **2021**, *10*, 559–570. (In Chinese)
35. Guillaume, R.; Ali, M.D.; Cyrille, E. Radar emitters classification and clustering with a scale mixture of Normal distributions. *IET Radar Sonar Navig.* **2018**, *13*, 128–138.
36. Salles, R.; Belloze, K.T.; Porto, F.; Gonzalez, P.H.; Ogasawara, E. Nonstationary time series transformation methods: An experimental review. *Knowl. Based Syst.* **2019**, *164*, 274–291. [CrossRef]
37. Esling, P.; Agon, C. Time-series data mining. *ACM Comput. Surv.* **2012**, *45*, 1–34. [CrossRef]
38. Aghabozorgi, S.; Shirkhorshidi, A.S.; Wah, T.Y. Time-series clustering—A decade review. *Inf. Syst.* **2015**, *53*, 16–38. [CrossRef]
39. Zolhavarieh, S.; Aghabozorgi, S.; Teh, Y.W. A Review of Subsequence Time Series Clustering. *Sci. World J.* **2014**, *2014*, 312521. [CrossRef]
40. Liao, T.W. Clustering of time series data—a survey. *Pattern Recognit.* **2005**, *38*, 1857–1874. [CrossRef]
41. Chamroukhi, F.; Mohammed, S.; Trabelsi, D.; Oukhellou, L.; Amirat, Y. Joint segmentation of multivariate time series with hidden process regression for human activity recognition. *Neurocomputing* **2013**, *120*, 633–644. [CrossRef]
42. Han, M.; Liu, X. Feature selection techniques with class separability for multivariate time series. *Neurocomputing* **2013**, *110*, 29–34. [CrossRef]
43. Tuncel, K.S.; Baydogan, M.G. Autoregressive forests for multivariate time series modeling. *Pattern Recognit.* **2018**, *73*, 202–215. [CrossRef]
44. Zhou, L.; Du, G.; Tao, D.; Chen, H.; Cheng, J.; Gong, L. Clustering Multivariate Time Series Data via Multi-Nonnegative Matrix Factorization in Multi-Relational Networks. *IEEE Access* **2018**, *6*, 74747–74761. [CrossRef]
45. González-Vidal, A.; Jiménez, F.; Gómez-Skarmeta, A.F. A methodology for energy multivariate time series forecasting in smart buildings based on feature selection. *Energy Build.* **2019**, *196*, 71–82. [CrossRef]
46. Li, H. Multivariate time series clustering based on common principal component analysis. *Neurocomputing* **2019**, *349*, 239–247. [CrossRef]
47. Jiménez, F.; Palma, J.; Sánchez, G.; Marín, D.; Francisco Palacios, M.D.; Lucía López, M.D. Feature selection based multivariate time series forecasting: An application to antibiotic resistance outbreaks prediction. *Artif. Intell. Med.* **2020**, *104*, 101818. [CrossRef] [PubMed]
48. Li, H.; Liu, Z. Multivariate time series clustering based on complex network. *Pattern Recognit.* **2021**, *115*, 107919. [CrossRef]
49. Sanhudo, L.; Rodrigues, J.; Filho, E.V. Multivariate time series clustering and forecasting for building energy analysis: Application to weather data quality control. *J. Build. Eng.* **2021**, *35*, 101996. [CrossRef]
50. Wiley, R. *ELINT: The Interception and Analysis of Radar Signals*; Artech House: Norwood, MA, USA, 2006; Volume 111. [CrossRef]
51. Ahmadzadeh, A.; Sinha, K.; Aydin, B.; Angryk, R.A. MVTS-Data Toolkit: A Python package for preprocessing multivariate time series data. *SoftwareX* **2020**, *12*, 100518. [CrossRef]

52. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A Search Space Odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2222–2232. [[CrossRef](#)] [[PubMed](#)]
53. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
54. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
55. Li, Y.; Ma, D.; Zhu, M.; Zeng, Z.; Wang, Y. Identification of significant factors in fatal-injury highway crashes using genetic algorithm and neural network. *Accid. Anal. Prev.* **2018**, *111*, 354–363. [[CrossRef](#)] [[PubMed](#)]
56. Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416. [[CrossRef](#)]
57. Feng, X.; Hu, X.; Liu, Y. Radar signal sorting algorithm of k-means clustering based on data field. In Proceedings of the 2017 3rd IEEE International Conference on Computer and Communications (ICCC), Chengdu, China, 3–16 December 2017; pp. 2262–2266. [[CrossRef](#)]
58. Yang, Z.; Wu, Z.; Yin, Z.; Quan, T.; Sun, H. Hybrid Radar Emitter Recognition Based on Rough k-Means Classifier and Relevance Vector Machine. *Sensors* **2013**, *13*, 848–864. [[CrossRef](#)] [[PubMed](#)]
59. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining—KDD'96, Portland, OR, USA, 2–4 August 1996; AAAI Press: Washington, DC, USA, 1996; pp. 226–231.
60. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.