

Article

Continued Electromagnetic Signal Classification Based on Vector Space Separation

Lu Jia ¹, Yan Zhao ², Shichuan Chen ^{1,*}  and Zhijin Zhao ²¹ National Key Laboratory of Electromagnetic Space Security, Jiaxing 314033, China² Telecommunication School, Hangzhou Dianzi University, Hangzhou 310018, China

* Correspondence: sicanier@sina.com

Abstract

Incremental electromagnetic signal classification is crucial in realistic wireless environments where new signal types continuously emerge and historical training data are often unavailable. This paper proposes a model-based incremental learning method driven by vector space separation to mitigate catastrophic forgetting without accessing old-task samples or requiring semantic information. We show that forgetting is largely caused by insufficient separation between old and new classes in the classifier weight space. To address this issue, we jointly introduce weight normalization, a cosine-similarity separation loss, and regularization, together with cross-entropy supervision for new classes. Based on these designs, we propose an incremental learning method based on vector space separation for electromagnetic signal classification, enabling the model to continually recognize modulation signals without requiring semantic information or access to raw data from previous tasks during incremental updates. Experiments on two simulated modulation datasets under multiple task sequences demonstrate that the proposed method consistently alleviates catastrophic forgetting and achieves stable incremental performance, outperforming baselines while avoiding data rehearsal.

Keywords: electromagnetic signal classification; vector space separation; catastrophic forgetting

1. Introduction

Electromagnetic signal recognition is a core technology underpinning radio monitoring, cognitive radio, and communication reconnaissance systems. In military scenarios, accurate identification of intercepted signals is a prerequisite for effective signal interception, analysis, and jamming. In civilian applications, electromagnetic signal recognition plays an important role in spectrum monitoring and management [1].

Traditional electromagnetic signal recognition methods can generally be categorized into two classes: likelihood-based (LB) methods [2] and feature-based (FB) methods [3]. Likelihood-based methods rely on the statistical modeling of received signals and construct decision rules by minimizing predefined cost functions. These approaches typically establish signal statistics and compare them with preset thresholds to perform signal identification. Representative methods include the generalized likelihood ratio test (GLRT) [4,5], the average likelihood ratio test (ALRT) [6,7], and the hybrid likelihood ratio test (HLRT) [8,9]. Feature-based electromagnetic signal recognition methods [10–12] first extract discriminative features from received signals and then perform classification using suitable classifiers.



Academic Editor: Giulio Antonini

Received: 27 December 2025

Revised: 27 January 2026

Accepted: 28 January 2026

Published: 30 January 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

The effectiveness of these methods is highly dependent on the quality and representativeness of the selected features. Extracted features can generally be grouped into three categories: time-domain features, transform-domain features, and higher-order cumulant (HOC) features.

Despite their effectiveness, traditional likelihood-based and feature-based methods often suffer from several limitations, including complex feature extraction procedures, high computational complexity, and limited scalability, which hinder their deployment in practical large-scale systems [13]. With the rapid development of artificial intelligence and big data technologies, deep learning has emerged as a promising alternative for electromagnetic signal recognition [14–22]. By directly learning representations from raw or minimally processed signal samples, deep neural networks can automatically extract discriminative features and achieve superior classification performance. Compared with traditional methods, deep learning-based approaches exhibit stronger adaptability, higher recognition accuracy, and better real-time performance. Currently, deep learning-based electromagnetic signal recognition methods mainly rely on three types of neural networks: convolutional neural networks (CNNs), recurrent neural networks (RNNs), and graph neural networks (GNNs). CNN-based approaches have been widely adopted due to their powerful feature extraction capability. For example, ref. [23] employed CNNs to classify received orthogonal frequency-division multiplexing (OFDM) signals. Additionally, ref. [24] introduced a feature-image-based classification method, which outperformed traditional constellation-image-based approaches. RNNs are particularly suitable for modeling sequential data. In [25], the performance of LSTM and bidirectional LSTM models was systematically compared. The integration of CNN and LSTM modules was shown to further enhance recognition accuracy, especially under low-SNR conditions [26]. In [27], an LSTM-based denoising autoencoder framework was proposed, achieving efficient radio classification on low-cost computing platforms. Attention-enhanced LSTM models were also investigated for radar signal identification in [28]. Graph neural networks provide a flexible framework for modeling structured data that cannot be naturally handled by CNNs or RNNs. In [29], time-series signals were mapped into graph representations using an adaptive visibility graph algorithm for classification. A graph convolutional network (GCN)-based electromagnetic signal recognition method was proposed in [30], where signals were converted into graph structures before classification. Furthermore, ref. [31] transformed constellation maps into graph-like feature representations and achieved higher classification accuracy than competing methods.

Although deep learning-based models can achieve high recognition accuracy under static conditions, practical electromagnetic environments are inherently dynamic, with new signal types continuously emerging. In such scenarios, recognition models must not only retain knowledge of previously learned signal classes but also adapt to newly encountered ones. This requirement highlights the importance of sustainable learning and motivates the adoption of incremental learning techniques. Incremental learning aims to enable models to learn new tasks while preserving performance on old tasks, typically under constraints of limited or no access to historical data. A key challenge in this field is the stability–plasticity dilemma, where learning new knowledge often leads to catastrophic forgetting of previously acquired knowledge [32].

Most existing incremental learning methods have been developed for image-based tasks, while research focusing on electromagnetic signals remains limited. In [33], a large-scale incremental learning framework based on constellation images was proposed, where a small subset of historical data was retained. However, converting signals into constellation representations may result in information loss, degrading recognition performance. In [34], representative samples from previous tasks were stored to mitigate forgetting, but the

memory requirement grows as the number of incremental tasks increases. The work in [35] maintained classification performance by selecting key samples and employing a multi-objective linear programming classifier to rapidly adapt to new categories.

Unlike the above data-rehearsal-based approaches, this paper focuses on model-based incremental learning and proposes a vector-space-separation method for electromagnetic signal classification, which alleviates catastrophic forgetting without storing historical samples. This setting is particularly suitable for electromagnetic signal recognition, where memory, storage, and data-access permissions are often strictly constrained. Moreover, many widely used incremental learning methods originally developed for image recognition still suffer from decision-boundary drift and a bias toward newly introduced classes, which can substantially degrade performance on previously learned modulation types. In contrast, the proposed method explicitly preserves geometric separation between old and new classes in the classifier weight space through weight normalization and a cosine-similarity-based separation loss, leading to improved incremental performance.

The main contributions of this paper are as follows:

- A vector-space-driven incremental learning framework for electromagnetic signal classification is proposed, which enables continuous learning of new modulation types without accessing historical data or requiring explicit old-task semantic information.
- A novel weight-space separation mechanism is introduced by jointly leveraging normalization and cosine similarity constraints, explicitly enforcing angular separation between old and new class weight vectors to suppress catastrophic forgetting at the classifier level.
- A unified incremental optimization objective is formulated by integrating similarity-based separation loss, regularization, and cross-entropy loss, ensuring both effective preservation of old knowledge and discriminative learning of new classes.
- Comprehensive experimental evaluation is conducted on two modulation datasets under diverse incremental task sequences, systematically analyzing the impact of task order, signal diversity, and initial task quality on incremental learning performance.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the proposed incremental learning method and model formulation. Section 4 reports experimental results on two simulated datasets. Finally, Section 5 concludes the paper.

2. Related Work

Focusing on incremental learning, existing studies can be broadly grouped into two paradigms: model-based methods and old-sample-based (rehearsal or generative replay) methods.

Model-based incremental learning. This paradigm preserves knowledge of previous tasks primarily by optimizing the recognition model itself, rather than storing historical data. In [36], the authors compared the effectiveness of using fixed feature extractors [37] versus fine-tuning strategies [38] for continual learning. Motivated by knowledge distillation [39], they further proposed a non-forgetting learning scheme to mitigate catastrophic forgetting. In [40], the authors argued that the imbalance between old and new classes is a key factor behind forgetting, and introduced cosine normalization, forgetting constraints, and inter-class separation to address this issue. Similarly, ref. [41] employed distillation to reduce performance degradation on old tasks and adopted weight alignment to rebalance old and new classes. Beyond distillation, ref. [42] proposed a network re-parameterization strategy that, when combined with elastic weight consolidation, improves incremental learning performance. Parameter-regularization approaches such as [43,44] preserve previous knowledge by constraining parameters deemed important

for earlier tasks. In addition, topology-aware methods [45,46] incorporate structural priors into neural networks and progressively reserve different feature representations for continual learning.

Old-sample-based incremental learning. This paradigm alleviates forgetting by retaining a subset of historical data (or their substitutes) and replaying them during training. In [47], a fixed memory budget was introduced to store a small set of representative exemplars, which are dynamically updated as new tasks arrive, thereby reducing catastrophic forgetting; subsequent work further scaled this idea for large settings [48]. In [49], the gradient episodic memory method formulates exemplar selection and retention as a constrained optimization problem, helping protect performance on old tasks while learning new ones. Another line of work replaces explicit storage with generative replay, where a generative model synthesizes samples from past tasks [50]. For example, GAN-based generators have been used to produce replay samples [51]. In [52], old-task data are approximated by samples generated from a stored generator, although the effectiveness largely depends on the generator's fidelity. More recently, ref. [53] proposed generating subnetworks from previous data to retain prior knowledge, which can further mitigate catastrophic forgetting.

3. Method

3.1. Overall Framework

In this paper, we propose an incremental learning approach based on vector-space separation. During incremental updates, the model is expected to learn new classes without requiring the original data or explicit identity information from previous tasks, thereby mitigating catastrophic forgetting while adapting to newly observed modulation types.

As discussed in [48], a common practice for class-incremental recognition with deep neural networks is to expand the classifier by adding new output neurons (i.e., enlarging the final fully connected layer) when new classes arrive. However, if no additional treatment is applied, the classifier tends to become biased toward newly learned classes, causing the model to forget previously learned knowledge. From a geometric perspective, this phenomenon can be attributed to an unconstrained reconfiguration of the classifier weight space: since the model is updated using only new-task samples, the newly added class weight vectors may drift toward directions that are not sufficiently separated from those of previously learned classes. As a result, the weight vectors of old and new classes can become highly similar or even overlap in the weight space, which blurs the decision boundaries and increases confusion between old and new classes, ultimately leading to severe performance degradation on previously learned modulation classes.

Let matrix $\mathbf{A}$ denote the weight parameters of the final fully connected layer after training on the first task. Suppose the first task contains three classes, and each class corresponds to a 512-dimensional weight vector. Then $\mathbf{A}$ can be written as

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,512} \\ a_{2,1} & a_{2,2} & \dots & a_{2,512} \\ a_{3,1} & a_{3,2} & \dots & a_{3,512} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}. \quad (1)$$

During continual learning, new classes are introduced and the classifier is correspondingly expanded. Let $\mathbf{B}$ denote the newly added weight vectors learned for the new task. If the new task contains two classes, then $\mathbf{B}$ is expressed as

$$\mathbf{B} = \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,512} \\ b_{2,1} & b_{2,2} & \dots & b_{2,512} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}. \quad (2)$$

At this stage, the model is updated using only new-task samples, while the classifier parameters for old classes (contained in **A**) are not sufficiently reinforced. Consequently, if no constraints are imposed, catastrophic forgetting may occur, as illustrated in Figure 1. Specifically, the weight vectors associated with old and new classes become imbalanced in the classifier weight space: the updated decision boundaries tend to drift toward new classes, which degrades discrimination performance on old classes.

Motivated by this observation, the core idea of our method is to explicitly separate the new weight vectors from the old ones in the classifier weight space, thereby reducing interference from new-task learning on previously learned knowledge. The overall workflow is illustrated in Figure 2. To realize vector-space separation, we integrate normalization and regularization operations into the classifier optimization, and employ cosine similarity together with cross-entropy loss to guide the learning of discriminative and well-separated class weight vectors.

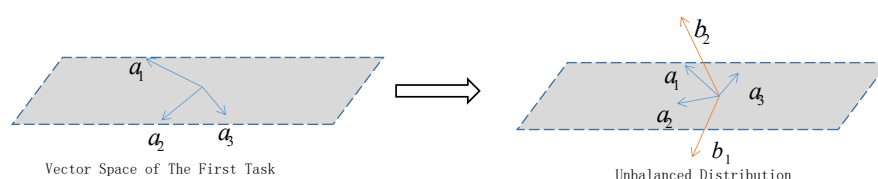


Figure 1. Illustration of catastrophic forgetting problem.

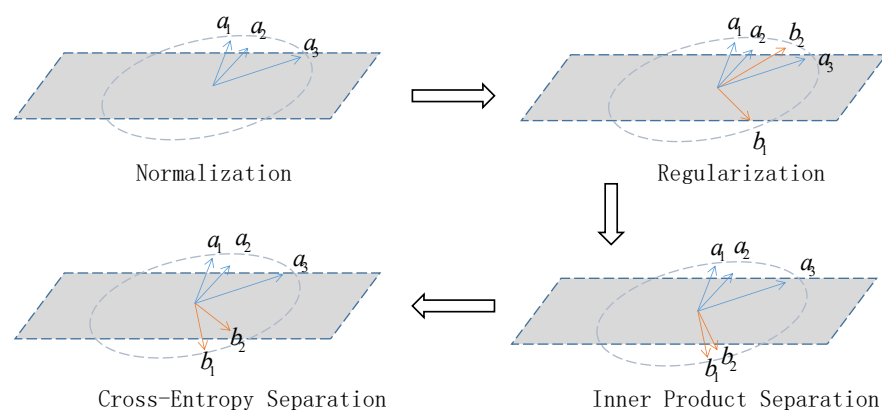


Figure 2. Overall workflow of the proposed method.

3.2. Spatial Separation Method

When training the model on the first task, the weight vectors of the final fully connected layer are first normalized to ensure compactness and stability in the classifier weight space. This normalization step helps prevent excessive dispersion of weight vectors, thereby facilitating subsequent incremental learning stages.

When a new task arrives, a new fully connected layer corresponding to the new classes is appended to the classifier of the original model, and the outputs are concatenated. During this incremental update, a similarity-based loss is introduced to explicitly separate the weight vectors associated with new and old tasks in the vector space. Meanwhile, regularization is applied to control the magnitude of newly learned weights, ensuring smooth adaptation and facilitating future incremental updates. The cross-entropy loss is used to discriminate among the classes within the new task, enabling effective learning of new knowledge.

To prevent weight dispersion in the vector space, the parameters of the final fully connected layer are normalized before training the model on the first task. This process is defined as

$$\tilde{a}_{i,j} = \frac{a_{i,j} - a_{\min}}{a_{\max} - a_{\min}}, \quad (3)$$

where $a_{i,j}$ and $\tilde{a}_{i,j}$ denote the (i, j) -th elements of the weight matrix before and after normalization, respectively, and $a_{\min}$ and $a_{\max}$ represent the minimum and maximum values of all elements in the weight matrix.

When a subsequent task is introduced, a new fully connected layer is added to the original model and concatenated with the existing classifier. The resulting weight matrix of the output layer can be expressed as

$$\mathbf{W} = (\mathbf{A}, \mathbf{B}), \quad (4)$$

where

$$\mathbf{A} = (a_1, a_2, \dots, a_m), \quad \mathbf{B} = (b_1, b_2, \dots, b_n), \quad (5)$$

with m and n denoting the numbers of old and new classes, respectively. During training, the parameters corresponding to $\mathbf{A}$ are fixed, while only the new-task weights $\mathbf{B}$ are updated.

To minimize interference between old and new tasks, cosine similarity is employed to measure the angular relationship between weight vectors of old and new classes:

$$\text{similarity} \langle \mathbf{a}, \mathbf{b} \rangle = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\|_2 \|\mathbf{b}\|_2} = \frac{\sum_{i=1}^n a_i b_i}{\sqrt{\sum_{i=1}^n a_i^2} \sqrt{\sum_{i=1}^n b_i^2}}, \quad (6)$$

Based on this measure, the similarity loss is defined as

$$L_w = \max_{i,j} [\text{similarity} \langle a_i, b_j \rangle], \quad (7)$$

which encourages maximal separation between old-task and new-task weight vectors in the classifier space.

To further balance the scale of old and new weights, an L_2 regularization term is introduced:

$$L_2 = \frac{1}{2\rho} \|\mathbf{B}\|_2^2, \quad (8)$$

where ρ denotes the number of training samples and $\mathbf{B}$ represents the collection of all weight vectors associated with the new task.

The discrimination among new classes is achieved using the standard cross-entropy loss:

$$L_{ce} = -\frac{1}{\rho} \sum_{i=1}^{\rho} \sum_{j=1}^n \mathcal{L}_j^{\text{train}} \log \hat{\mathcal{L}}_j^{\text{train}}, \quad (9)$$

where $\mathcal{L}_j^{\text{train}}$ and $\hat{\mathcal{L}}_j^{\text{train}}$ denote the ground-truth label and the predicted probability for the j -th class, respectively.

In summary, the total training objective is composed of three terms: the similarity loss for separating old and new class weights, the regularization term for stabilizing incremental updates, and the cross-entropy loss for new-class discrimination. The overall loss function is formulated as

$$L_{\text{total}} = \lambda L_w + \eta L_2 + L_{ce}, \quad (10)$$

where λ controls the contribution of the similarity loss and η is the regularization coefficient.

3.3. Incremental Process

The overall incremental learning framework is illustrated in Figure 3. The proposed classifier consists of two main components: a feature extraction module and a concatenated

fully connected classifier. The objective is to develop a continual learning strategy that enables robust signal classification across multiple incremental tasks.

Before the initial training stage, the weight parameters of the final fully connected layer in the base network are normalized. An initial classification model is then trained using a dataset X^M containing samples from M known classes. This process can be formulated as

$$\zeta(M) = D(X^M), \quad (11)$$

where $D()$ denotes the training operation of the classifier network and $\zeta(M)$ represents the model obtained after learning the first M classes.

During subsequent incremental stages, when a new task containing N unseen classes arrives, an additional fully connected layer is appended to the classifier of the previously trained model. The newly added and existing fully connected layers are concatenated to construct an updated classifier, denoted as $\zeta(M)'$. During training, the feature extraction layers and the classifier weights associated with old classes are frozen, while only the newly added classifier parameters are optimized using the dataset X^N . The incremental learning process can be expressed as

$$\zeta(M+N) = D(\zeta(M)', X^N). \quad (12)$$

where $\zeta(M+N)$ denotes the updated model after incorporating the new classes. By repeating this procedure, the model achieves continual learning over multiple incremental tasks without retraining from scratch.

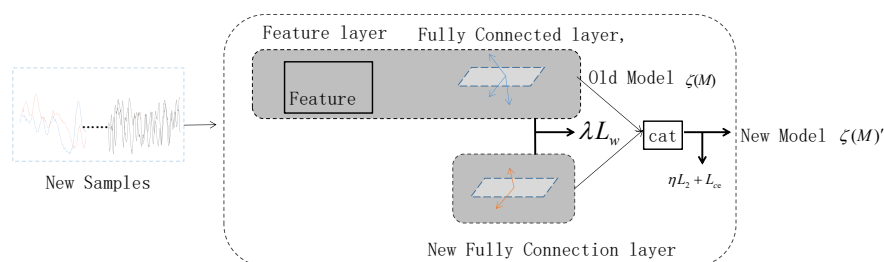


Figure 3. Overall incremental learning framework.

3.4. Classification Network

In this paper, a deep neural network classifier is employed to categorize the received signals. The input signal sequence is represented as

$$x(n) = [x_1, x_2, \dots, x_{num}], \quad (13)$$

where num represents the number of signal samples. For each sample, we use the in-phase and quadrature components as two input channels to the network. Specifically, the I channel corresponds to the real part of $x(n)$, and the Q channel corresponds to the imaginary part of $x(n)$, which can be expressed as

$$\begin{aligned} I &= \text{Re}\{x(n)\}, \\ Q &= \text{Im}\{x(n)\}. \end{aligned} \quad (14)$$

The overall architecture of the proposed electromagnetic signal classification network is illustrated in Figure 4. In the figure, Conv denotes a convolutional layer, where the number following it indicates the number of convolution kernels, and “/2” represents a down-sampling factor of 2, meaning that the output resolution is reduced to half of the input. Maxpool and Avgpool denote maximum pooling and average pooling operations,

respectively. Fc represents a fully connected layer, and Softmax denotes the Softmax layer. The convolutional layers connected by dotted lines are used to ensure consistent feature dimensions across different branches. Pooling operations are employed to control the spatial size of feature maps, and finally, the Softmax layer converts the network outputs into a probability distribution over signal classes.

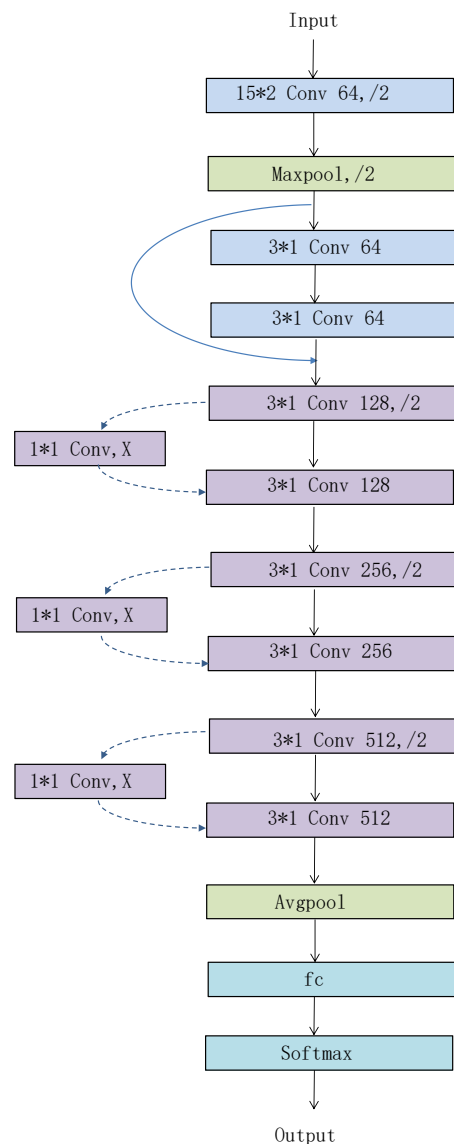


Figure 4. Structure of the constructed residual classification network.

4. Simulations

4.1. Datasets

In this paper, we evaluate the effectiveness of the proposed method on two simulation-generated datasets.

- **SignalData1.** SignalData1 contains a total of 115,200 signal samples from 12 modulation types: BPSK, QPSK, 8PSK, OQPSK, 2FSK, 4FSK, 8FSK, 16QAM, 32QAM, 64QAM, 4PAM, and 8PAM. Each modulation type has 9600 samples, and each sample has a length of 512. The SNR ranges from 16 to 30 dB with a step size of 2 dB, resulting in 8 SNR levels in total. The dataset is randomly split into a training set and a test set with a ratio of 5:1. For each modulation type and each SNR level, the training set includes 1000 samples, while the test set includes 200 samples. The test set is used for performance evaluation.

- **SignalData2.** SignalData2 also contains 115,200 signal samples and covers 12 modulation types: 2FSK, 4FSK, 8FSK, 16QAM, 32QAM, 64QAM, 128QAM, AQPSK, BPSK, MSK, OQPSK, and QPSK. Each modulation type consists of 9600 samples, with a sample length of 512. The SNR ranges from 16 to 30 dB in increments of 2 dB (8 SNR levels). The dataset is randomly divided into training and test sets with a 5:1 split. Specifically, for each modulation type at each SNR level, 1000 samples are used for training and 200 samples are used for testing. The test set is used to evaluate the performance of the methods.

4.2. Baselines and Setup

To evaluate the effectiveness of the proposed method, we compare it with four representative baseline approaches, including Fine-tuning, Feature Extraction, Learning without Forgetting (LWF), and Joint Training. The overall workflows of these methods are illustrated in Figure 5.

- **Fine-tuning** [47]. Starting from a trained model on previous tasks, a new fully connected layer corresponding to the new classes is appended to the classifier and concatenated with the existing output layer. Fine-tuning is performed using only samples from the new task. During training, the parameters associated with old tasks are updated with a smaller learning rate to mitigate excessive forgetting, while the newly added classifier weights are optimized normally. This baseline reflects naïve incremental adaptation and typically exhibits decision-boundary drift.
- **Feature Extraction** [47]. This method follows a similar pipeline to Fine-tuning, but differs in that the parameters of the feature extraction layers learned from old tasks are completely frozen during incremental training. The frozen layers act as a fixed feature extractor, and only the newly added classifier layers are trained to adapt to the new task. This baseline represents fixed-representation learning with limited plasticity for new classes.
- **LWF** [36]. Unlike Fine-tuning, Learning without Forgetting requires saving the trained classification model after each incremental task. When a new task arrives, the parameters of the previous model are loaded, and the final fully connected layer is expanded to accommodate new classes. Knowledge distillation is applied by aligning the predictions of the new model with those produced by the old model on the new-task data. Meanwhile, the classifier weights corresponding to old tasks are fine-tuned using a small learning rate, and the cross-entropy loss is employed to learn the new task, thereby preserving knowledge of previous tasks as much as possible. This baseline is a representative distillation-based continual learning method.
- **Joint Training** [54]. For each incremental stage, Joint Training retrains the classification network using the combined datasets of both old and new tasks, while modifying the final fully connected layer to match the total number of classes. This approach can achieve strong performance but requires access to all historical data, leading to high storage costs and increased training time. This baseline is commonly treated as an oracle upper bound under full data access.

For all incremental tasks and baseline methods, the same training strategy and experimental environment are adopted to ensure a fair comparison. The Adam optimizer is used for network optimization. The initial learning rate is set to 0.01, with a learning rate decay factor of 0.8. The batch size is fixed at 128. For methods involving fine-tuning of previously learned layers, a smaller learning rate of 0.0001 is applied. For the proposed method, the hyperparameters are set to $\lambda = 0.1$ and $\eta = 0.3$ in all experiments. We implemented the methods using the PyTorch 2.6.0, and the hardware used includes an Intel(R) Core(TM) i9-12900HX CPU and an NVIDIA GeForce RTX 3080Ti Laptop GPU.

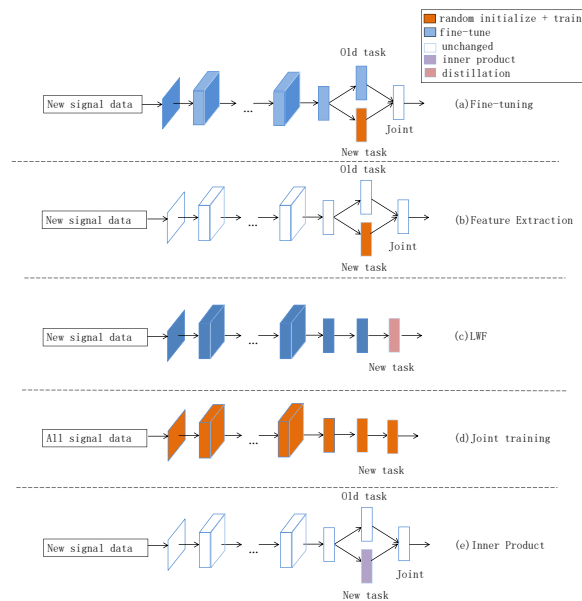


Figure 5. The overall workflows of baseline methods.

4.3. Simulation Results

4.3.1. Performance Comparison

We construct two different recognition task sequences for the two signal datasets. In these sequences, both the type and the number of classes introduced at each stage vary, which leads to changes in the scale and composition of the training data available at each update, the proportion of newly introduced classes, and the prevailing SNR conditions. As a result, the incremental process naturally incorporates practical factors such as varying data availability, class imbalance, and non-stationary noise. The average classification accuracies of each task sequence under the five methods are reported in Tables 1 and 2, respectively. The corresponding trends of average accuracy versus the number of classes are illustrated in Figure 6a,b. As can be observed from Tables 1 and 2, and Figure 6, Joint training consistently achieves the best performance. This is mainly because Joint training optimizes the model using both old-task and new-task data, enabling the classifier to learn discriminative representations for all classes with minimal distribution mismatch. However, this advantage comes at the cost of storing all historical samples and incurring higher training complexity.

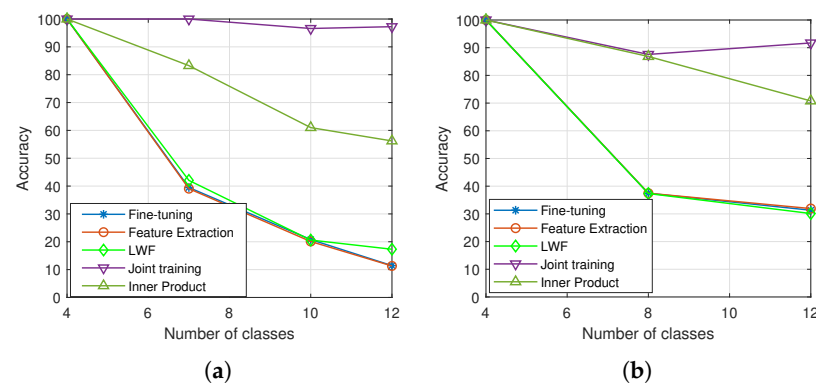
Compared with Joint training, the other incremental paradigms exhibit different degrees of catastrophic forgetting. For Fine-tuning, although the parameters associated with old tasks are updated with a smaller learning rate while new-task layers are learned with a larger learning rate, the optimization driven by new-task data still shifts decision boundaries toward new classes, leading to substantial degradation on old classes. For Feature extraction, the feature extractor learned from previous tasks is frozen and only the classifier for new classes is trained; consequently, the learning capacity for new tasks is constrained by the previously learned representation, and the classifier tends to bias toward new classes as well, which also results in forgetting. Therefore, Fine-tuning and Feature extraction generally show similar performance. By introducing a distillation loss, LWF partially preserves the outputs of the old model and thus alleviates forgetting to some extent, yielding better performance than Fine-tuning and Feature extraction.

Table 1. Task Sequence 1 and Recognition Rate.

Task	Signal Type	Fine-Tuning	Feature Extraction	LWF	Joint Training	Inner Product
1	BPSK, QPSK, 8PSK, OQPSK	100	100	100	100	100
2	2FSK, 4FSK, 8FSK	39.48	39.07	42.05	100.00	83.26
3	16QAM, 32QAM, 64QAM	20.79	20.09	20.61	96.60	61
4	4PAM, 8PAM	11.43	11.31	17.31	97.25	56.21

Table 2. Task Sequence 2 and Recognition Rate.

Task	Signal Type	Fine-Tuning	Feature Extraction	LWF	Joint Training	Inner Product
1	16QAM, 32QAM, 64QAM, 128QAM	100	100	100	100	100
2	BPSK, QPSK, AQPSK, OQPSK	37.44	37.46	37.34	87.57	86.83
3	2FSK, 4FSK, 8FSK, MSK	31.32	31.92	30.13	91.71	70.78

**Figure 6.** The recognition performance of five methods under different task sequences. (a) sequence1, (b) sequence2.

To further examine class-wise behaviors, the confusion matrices of different methods are presented in Figure 7. These class-level patterns are consistent with the average accuracy trends reported in Table 1 for Sequence 1, where Fine-tuning and Feature Extraction show a pronounced performance drop as the number of classes increases. In Sequence 1, as shown in Figure 7a,b, Fine-tuning and Feature extraction exhibit a clear bias toward the new tasks, causing severe misclassification of old-task classes. As shown in Figure 7c, LWF improves the recognition performance of some old-task classes (e.g., 2FSK) compared with Fine-tuning and Feature extraction, indicating that distillation provides additional retention of old knowledge, which is reflected by the moderate improvement in the average accuracies reported in Table 1. As shown in Figure 7d, Joint training achieves high recognition rates across almost all modulation types and therefore provides the best overall classification performance, in agreement with its consistently highest average accuracy values in Table 1. As shown in Figure 7e, the proposed method achieves non-trivial recognition accuracy on most classes and substantially mitigates catastrophic forgetting; only a few classes (e.g., OQPSK, 16QAM, and 32QAM) remain relatively challenging.

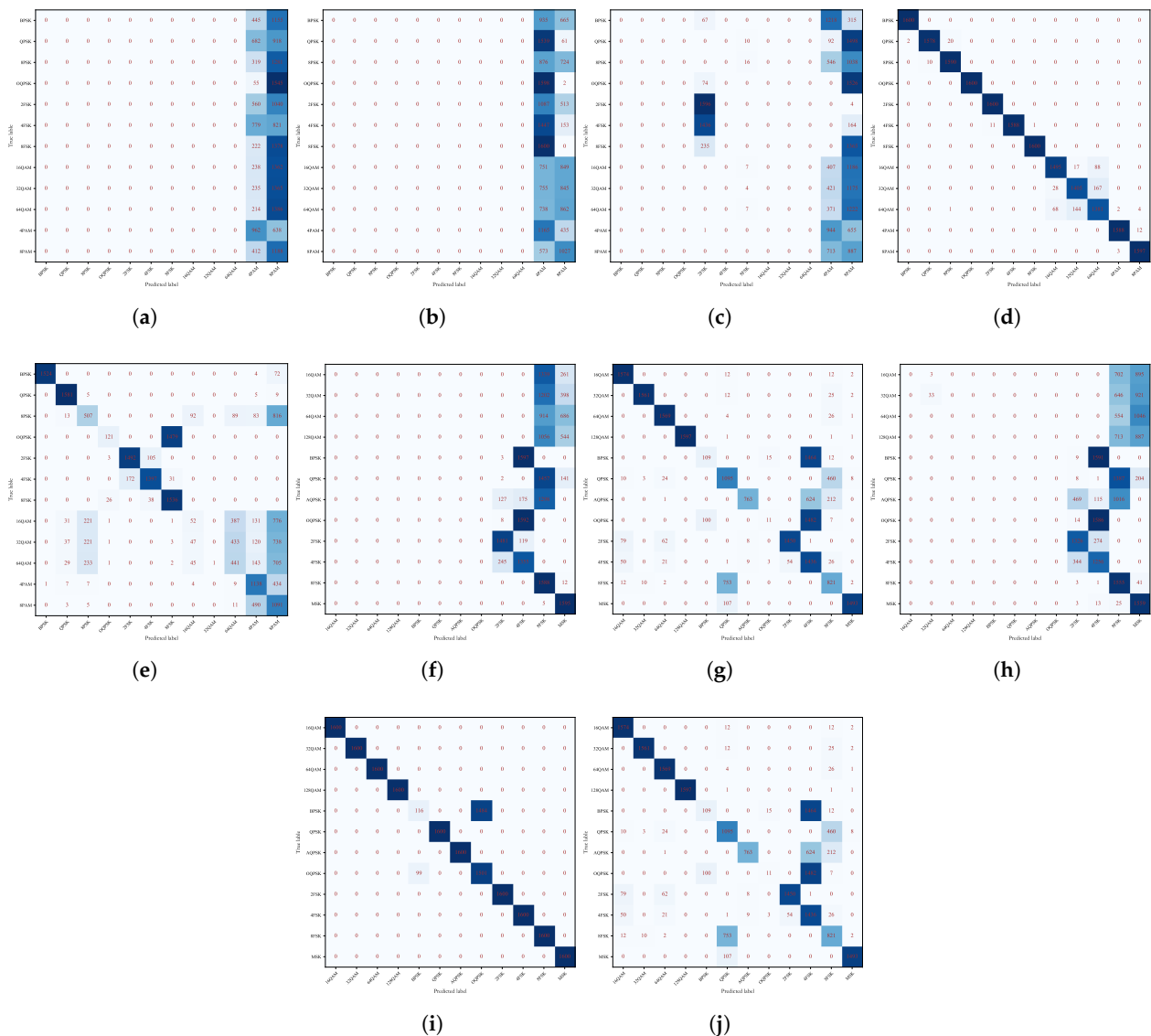


Figure 7. Confusion matrices for five methods on sequence1. (a) Fine-tuning, (b) Feature Extraction, (c) LWF, (d) Joint Training, (e) Inner Product. Confusion matrices for five methods on sequence2: (f) Fine-tuning, (g) Feature Extraction, (h) LWF, (i) Joint Training, (j) Inner Product.

In Sequence 2, Figure 7f–h show that Fine-tuning, Feature extraction, and LWF still suffer from catastrophic forgetting, with performance on old classes dropping markedly after learning new tasks. In contrast, as shown in Figure 7j, the proposed method maintains strong recognition capability for most modulation types; except for BPSK and OQPSK, the remaining classes achieve relatively high recognition rates, which is consistent with its superior and more stable average performance in Table 2, demonstrating effective resistance to catastrophic forgetting.

4.3.2. Influence of the Incremental Process

To investigate the influence of different task sequences on incremental learning performance, we conduct additional experiments on both SignalData1 and SignalData2. Specifically, three extra task sequences are designed for each dataset. For SignalData1, task sequences 3, 4, and 5 are added, while for SignalData2, task sequences 6, 7, and 8 are considered. The detailed configurations of all task sequences are summarized in Table 3.

The relationship between the average classification accuracy and the number of classes is illustrated in Figure 8.

Table 3. Task Sequence Setting.

Task	Sequence3	Sequence4	Sequence5	Sequence6	Sequence7	Sequence8
1	16QAM, 32QAM, 64QAM	BPSK, 2FSK, 16QAM	QPSK, 4FSK, 32QAM, 4PAM	BPSK, QPSK, AQPSK, OQPSK	4FSK, 32QAM, QPSK	2FSK, 16QAM, BPSK
2	4PAM, 8PAM	QPSK, 4FSK, 32QAM, 4PAM	BPSK, 2FSK, 16QAM	16QAM, 32QAM, 64QAM, 128QAM	8FSK, 16QAM, AQPSK	4FSK, 32QAM, QPSK
3	BPSK, QPSK, 8PSK, OQPSK	8PSK, 8FSK, 64QAM	8PSK, 8FSK, 64QAM	2FSK, 4FSK, 8FSK, MSK	2FSK, 64QAM, BPSK	8FSK, 64QAM, AQPSK
4	2FSK, 4FSK, 8FSK	OQPSK, 8PAM	OQPSK, 8PAM		MSK, 128QAM, OQPSK	MSK, 128QAM, OQPSK

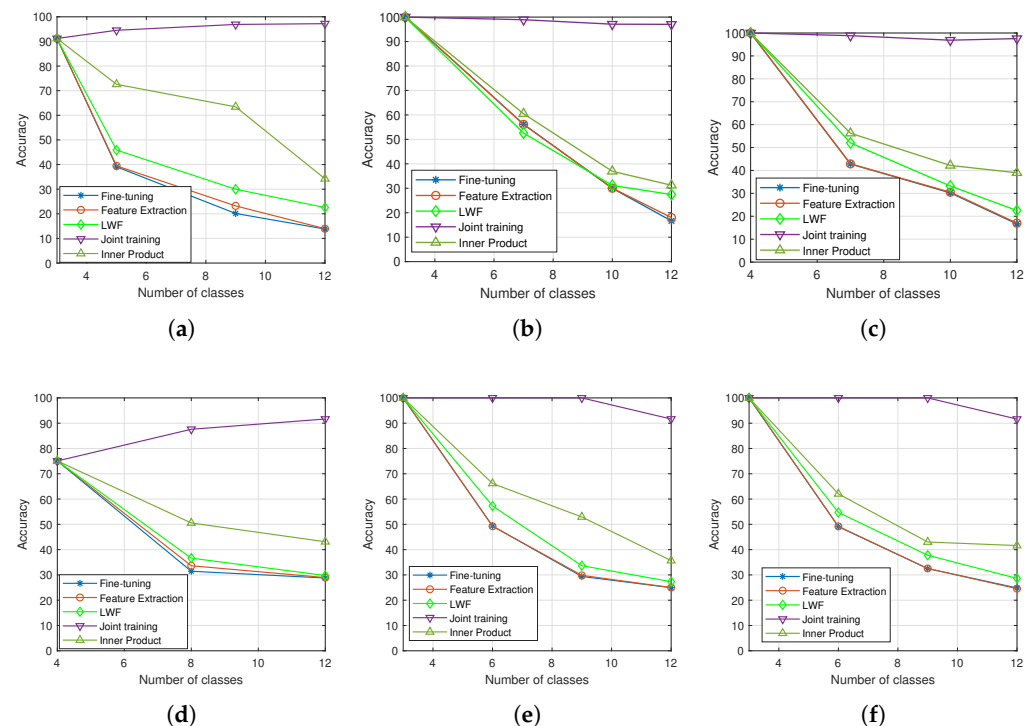


Figure 8. The recognition performance of five methods corresponding to the task sequence. (a) sequence3, (b) sequence4, (c) sequence5, (d) sequence6, (e) sequence7, (f) sequence8.

For SignalData1, task sequences 1 and 3 are composed of tasks with similar signal patterns. By comparing Figure 6a and Figure 8a, it can be observed that different task sequences have only a minor impact on the performance of each baseline method. For the proposed method, task sequence 1 yields a higher overall average recognition rate, and achieves the best final average accuracy after three incremental stages. In contrast, task sequences 4 and 5 include tasks with more diverse signal patterns, and all 12 signal types are learned through three incremental processes. As shown in Figure 8b,c, different task sequences lead to noticeable performance differences for the baseline methods. For the proposed method, task sequence 5 achieves the best performance after three incremental updates. By comparing Figure 6a with Figure 8a, as well as Figure 8b with Figure 8c, it can be concluded that both the number of classes in the initial task and its learning quality have a significant impact on subsequent incremental learning. Furthermore, comparisons between Figure 6a and Figure 8c, as well as Figure 8a and Figure 8b, indicate that when new and old tasks involve different signal styles, the model is more likely to maintain stable and effective continual learning behavior.

For SignalData2, task sequences 2 and 6 consist of tasks with similar signal patterns. As shown by comparing Figure 6b and Figure 8d, the choice of task sequence again has only a limited influence on the baseline methods, and the proposed method achieves better

overall performance in task sequence 2. Task sequences 7 and 8, however, contain tasks with more diverse signal types, and all 12 signal patterns are incrementally learned within three stages. As illustrated in Figure 8e,f, task sequence 8 yields the best incremental learning performance for the proposed method. The comparison between Figure 6b and Figure 8d further confirms that the learning quality of the initial task plays a crucial role in determining subsequent performance. Meanwhile, the comparison between Figure 8e and Figure 8f suggests that when new and old tasks share similar signal styles, the impact of task sequence variation on the proposed method becomes relatively small.

In summary, different task sequences can lead to distinct incremental learning outcomes. For the proposed method, incremental scenarios in which new and old tasks contain diverse signal types are more favorable for maintaining stable continual learning performance. In addition, the learning quality of the initial task significantly influences subsequent learning stages. Increasing the number of classes in the early tasks can enhance class discrimination, thereby improving the final average classification accuracy of the model.

5. Conclusions

In this paper, we propose an incremental learning method based on vector space separation for electromagnetic signal classification. The proposed approach enables the model to continually recognize modulation signals without requiring semantic information or access to raw data from previous tasks during incremental updates. Our analysis indicates that catastrophic forgetting largely arises from insufficient separation between old and new classes in the classifier weight space. To address this issue, we incorporate weight normalization and a similarity-based loss to explicitly enlarge the separation between old-class and new-class weight vectors, thereby reducing inter-task interference. We further investigate key factors that influence incremental performance, including task sequence design and the learning quality of the initial task. Experiments on two simulated datasets demonstrate that the proposed method effectively mitigates catastrophic forgetting and achieves stable incremental learning performance. In future work, we will extend the proposed framework to more diverse datasets and more challenging low-SNR and real-world scenarios, and explore more robust continual learning strategies.

Author Contributions: Conceptualization, Z.Z.; methodology, S.C.; validation, Y.Z.; writing—original draft, Y.Z.; Writing—review and editing, L.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: This research was supported by Starlit South Lake Leading Elite Program.

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

References

1. Alarabi, A.; Alkishriwo, O.A. Modulation classification based on statistical features and artificial neural network. In Proceedings of the 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering MI-STA, Tripoli, Libya, 25–27 May 2021; pp. 748–751.
2. Huang, S.; Yao, Y.; Xiao, Y.; Feng, Z. Cumulant based maximum likelihood classification for overlapped signals. *Electron. Lett.* **2016**, *52*, 1761–1763. [[CrossRef](#)]
3. Dobre, O.A.; Abdi, A.; Bar-Ness, Y.; Su, W. Survey of automatic modulation classification techniques: Classical approaches and new trends. *IET Commun.* **2007**, *1*, 137–156. [[CrossRef](#)]

4. Boiteau, D.; Le Martret, C. A general maximum likelihood framework for modulation classification. In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181), Seattle, WA, USA, 15 May 1998; Volume 4, pp. 2165–2168.
5. Hong, L.; Ho, K.C. Antenna array likelihood modulation classifier for BPSK and QPSK signals. In Proceedings of the Milcom, Anaheim, CA, USA, 7–10 October 2002.
6. Kim, K.; Polydoros, A. Digital modulation classification: The BPSK versus QPSK case. In Proceedings of the MILCOM 88, 21st Century Military Communications—What's Possible? Conference Record. Military Communications Conference, San Diego, CA, USA, 23–26 October 1988; Volume 2, pp. 431–436.
7. Sills, J. Maximum-likelihood modulation classification for PSK/QAM. In Proceedings of the MILCOM 1999. IEEE Military Communications. Conference Proceedings (Cat. No.99CH36341), Atlantic City, NJ, USA, 31 October–3 November 1999; Volume 1, pp. 217–220.
8. Hong, L.; Ho, K. BPSK and QPSK modulation classification with unknown signal level. In Proceedings of the MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority (Cat. No.00CH37155), Los Angeles, CA, USA, 22–25 October 2000; Volume 2, pp. 976–980.
9. Hong, L.; Ho, K. Modulation classification of BPSK and QPSK signals using a two element antenna array receiver. In Proceedings of the 2001 MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force (Cat. No.01CH37277), McLean, VA, USA, 28–31 October 2001; Volume 1, pp. 118–122.
10. Liedtke, F.F. Computer simulation of an automatic classification procedure for digitally modulated communication signals with unknown parameters. *Signal Process.* **1984**, *6*, 311–323. [\[CrossRef\]](#)
11. Wang, L.X.; Ren, Y.J.; Zhang, R.H. Algorithm of digital modulation recognition based on support vector machines. In Proceedings of the 2009 International Conference on Machine Learning and Cybernetics, Baoding, China, 12–15 July 2009; Volume 2, pp. 980–983.
12. Reichert, J. Automatic classification of communication signals using higher order statistics. In Proceedings of the IEEE International Conference on Acoustics, San Francisco, CA, USA, 23–26 March 1992; pp. 221–224.
13. Meng, F.; Chen, P.; Wu, L.; Wang, X. Automatic Modulation Classification: A Deep Learning Enabled Approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10760–10772. [\[CrossRef\]](#)
14. Zheng, S.; Zhou, X.; Zhang, L.; Qi, P.; Qiu, K.; Zhu, J.; Yang, X. Toward Next-Generation Signal Intelligence: A Hybrid Knowledge and Data-Driven Deep Learning Framework for Radio Signal Classification. *IEEE Trans. Cogn. Commun. Netw.* **2023**, *9*, 564–579. [\[CrossRef\]](#)
15. Qiu, K.; Zheng, S.; Zhang, L.; Lou, C.; Yang, X. DeepSIG: A Hybrid Heterogeneous Deep Learning Framework for Radio Signal Classification. *IEEE Trans. Wirel. Commun.* **2024**, *23*, 775–788. [\[CrossRef\]](#)
16. Chen, J.; Zhao, Z.; Zheng, S.; Zhang, L.; Qiu, K.; Yang, X. PCSP: A Novel Class Discovery Algorithm for Radio Signal Classification. *IEEE Trans. Cogn. Commun. Netw.* **2024**, *10*, 1190–1203. [\[CrossRef\]](#)
17. Zhang, D.; Yuan, Q.; Meng, L.; Xia, R.; Liu, W.; Qin, C. Reinforcement learning for single-agent to multi-agent systems: From basic theory to industrial application progress, a survey. *Artif. Intell. Rev.* **2025**, *59*, 46. [\[CrossRef\]](#)
18. Liang, L.; Tian, Z.; Huang, H.; Li, X.; Yin, Z.; Zhang, D.; Zhang, N.; Zhai, W. Heterogeneous Secure Transmissions in IRS-Assisted NOMA Communications: CO-GNN Approach. *IEEE Internet Things J.* **2025**, *12*, 34113–34125. [\[CrossRef\]](#)
19. Meng, L.L.; Si, X.J. An improved algorithm of modulation classification for digital communication signals based on wavelet transform. In Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition, Beijing, China, 2–4 November 2007; Volume 3, pp. 1226–1231.
20. Azzouz, E.E.; Nandi, A.K. Automatic identification of digital modulation types. *Signal Process.* **1995**, *47*, 55–69. [\[CrossRef\]](#)
21. Huang, S.; Jiang, Y.; Gao, Y.; Feng, Z.; Zhang, P. Automatic modulation classification using contrastive fully convolutional network. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 1044–1047. [\[CrossRef\]](#)
22. Chen, D.; Xiong, G.; Huang, B.; Wang, L. Modulation Signal Recognition Based on Feature Fusion and Residual CNN in Fractal Noise Background. In Proceedings of the 2021 IEEE 6th International Conference on Signal and Image Processing (ICSIP), Nanjing, China, 22–24 October 2021; pp. 687–691.
23. Song, G.; Jang, M.; Yoon, D. Cnn-based modulation classification for ofdm signal. In Proceedings of the 2021 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 20–22 October 2021; pp. 1326–1328.
24. Lee, J.H.; Kim, K.Y.; Shin, Y. Feature image-based automatic modulation classification method using CNN algorithm. In Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), Okinawa, Japan, 11–13 February 2019; pp. 1–4.
25. Karahan, S.N.; Kalaycioğlu, A. Deep Learning Based Automatic Modulation Classification With Long-Short Term Memory Networks. In Proceedings of the 2020 28th Signal Processing and Communications Applications Conference (SIU), Gaziantep, Turkey, 5–7 October 2020; pp. 1–4.

26. Zhou, Q.; Jing, X.; He, Y.; Cui, Y.; Kadoch, M.; Cheriet, M. LSTM-based Automatic Modulation Classification. In Proceedings of the 2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Paris, France, 27–29 October 2020; pp. 1–4.
27. Ke, Z.; Vikalo, H. Real-Time Radio Modulation Classification With An LSTM Auto-Encoder. In Proceedings of the ICASSP 2021—2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 4935–4939.
28. Wei, S.; Qu, Q.; Zeng, X.; Liang, J.; Shi, J.; Zhang, X. Self-Attention Bi-LSTM Networks for Radar Signal Modulation Recognition. *IEEE Trans. Microw. Theory Tech.* **2021**, *69*, 5160–5172. [\[CrossRef\]](#)
29. Xuan, Q.; Zhou, J.; Qiu, K.; Chen, Z.; Xu, D.; Zheng, S.; Yang, X. AvgNet: Adaptive Visibility Graph Neural Network and Its Application in Modulation Classification. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 1516–1526. [\[CrossRef\]](#)
30. Liu, Y.; Liu, Y.; Yang, C. Modulation Recognition With Graph Convolutional Network. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 624–627. [\[CrossRef\]](#)
31. Ghasemzadeh, P.; Hempel, M.; Sharif, H. A Robust Graph Convolutional Neural Network-Based Classifier for Automatic Modulation Recognition. In Proceedings of the 2022 International Wireless Communications and Mobile Computing (IWCMC), Dubrovnik, Croatia, 30 May–3 June 2022; pp. 907–912.
32. De Lange, M.; Aljundi, R.; Masana, M.; Parisot, S.; Jia, X.; Leonardis, A.; Slabaugh, G.; Tuytelaars, T. A continual learning survey: Defying forgetting in classification tasks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 3366–3385. [\[CrossRef\]](#)
33. Morehouse, T.; Rahimi, N.; Shao, M.; Zhou, R. Baseband Modulation Classification using Incremental Learning. In Proceedings of the 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), Springfield, MA, USA, 9–12 August 2020; pp. 225–228.
34. Yang, Y.; Chen, M.; Wang, X.; Ma, P. Modulation Recognition based on Incremental Deep Learning. In Proceedings of the 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Harbin, China, 25–27 December 2020; pp. 1701–1705.
35. Zhou, H.; Bai, J.; Niu, L.; Xu, J.; Xiao, Z.; Zheng, S.; Jiao, L.; Yang, X. Electromagnetic Signal Classification Based on Class Exemplar Selection and Multi-Objective Linear Programming. *Remote Sens.* **2022**, *14*, 1177. [\[CrossRef\]](#)
36. Li, Z.; Hoiem, D. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 2935–2947. [\[CrossRef\]](#)
37. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. In Proceedings of the International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 647–655.
38. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
39. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531. [\[CrossRef\]](#)
40. Hou, S.; Pan, X.; Loy, C.C.; Wang, Z.; Lin, D. Learning a unified classifier incrementally via rebalancing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 831–839.
41. Zhao, B.; Xiao, X.; Gan, G.; Zhang, B.; Xia, S.T. Maintaining discrimination and fairness in class incremental learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13208–13217.
42. Liu, X.; Masana, M.; Herranz, L.; Van de Weijer, J.; Lopez, A.M.; Bagdanov, A.D. Rotate your networks: Better weight consolidation and less catastrophic forgetting. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 2262–2268.
43. Rannen, A.; Aljundi, R.; Blaschko, M.B.; Tuytelaars, T. Encoder based lifelong learning. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1320–1328.
44. Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A.A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. Overcoming catastrophic forgetting in neural networks. *Proc. Natl. Acad. Sci. USA* **2017**, *114*, 3521–3526. [\[CrossRef\]](#)
45. Tao, X.; Chang, X.; Hong, X.; Wei, X.; Gong, Y. Topology-preserving class-incremental learning. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020. Proceedings, Part XIX 16. pp. 254–270.
46. Zhang, C.; Song, N.; Lin, G.; Zheng, Y.; Pan, P.; Xu, Y. Few-shot incremental learning with continually evolved classifiers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 12455–12464.
47. Rebuffi, S.A.; Kolesnikov, A.; Sperl, G.; Lampert, C.H. icarl: Incremental classifier and representation learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2001–2010.
48. Wu, Y.; Chen, Y.; Wang, L.; Ye, Y.; Liu, Z.; Guo, Y.; Fu, Y. Large scale incremental learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 374–382.

49. Aljundi, R.; Lin, M.; Goujaud, B.; Bengio, Y. Gradient based sample selection for online continual learning. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 11817–11826.
50. Nguyen, C.V.; Li, Y.; Bui, T.D.; Turner, R.E. Variational continual learning. *arXiv* **2017**, arXiv:1710.10628.
51. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative adversarial networks: An overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [[CrossRef](#)]
52. Shin, H.; Lee, J.K.; Kim, J.; Kim, J. Continual learning with deep generative replay. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 2990–2999.
53. Ammour, N. Memory Using Data Generator in Continual Learning for Remote Sensing Scene Classification. In Proceedings of the 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, Brussels, Belgium, 11–16 July 2021; pp. 4924–4927.
54. Castro, F.M.; Marín-Jiménez, M.J.; Guil, N.; Schmid, C.; Alahari, K. End-to-end incremental learning. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 233–248.

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.