electronics

MDPI

Article

Leveraging Incremental Learning for Dynamic Modulation Recognition

Song Ma [1], Lin Zhang [2,*], Zhangli Song [2], Wei Yu [1] and Tian Liu [1,*]

[1] Southwest China Institute of Electronic Technology, Chengdu 610036, China; ma_song@139.com (S.M.); yuwei2050748@sina.com (W.Y.)
[2] National Key Laboratory of Wireless Communications, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China; 202221220234@std.uestc.edu.cn
* Correspondence: linzhang1913@gmail.com (L.Z.); liutian139@139.com (T.L.)

Abstract: Modulation recognition is an important technology used to correctly identify the modulation modes of wireless signals and is widely used in cooperative and confrontational scenarios. Traditional modulation-recognition algorithms require the assistance of expert experiences, which constrains their applications. With the rapid development of artificial intelligence in recent years, deep learning (DL) is widely advocated for intelligent modulation recognition. Typically, DL-based modulation-recognition algorithms implicitly assume a relatively static scenario in which the signal samples of all the modulation modes can be completely collected in advance. In practical situations, the radio environment is quite dynamic and the signal samples with new modulation modes may appear sequentially, in which the current DL-based modulation-recognition algorithms may require unacceptable time and computing resource consumption to re-train the optimal model from scratch. In this study, we leveraged incremental learning (IL) and designed a novel IL-based modulation-recognition algorithm that consists of an initial stage and multiple incremental stages. The main novelty of the proposed algorithm lies in the new loss function design in each incremental stage, which combines the distillation loss of recognizing old modulation modes and the cross-entropy loss of recognizing new modulation modes. With the proposed algorithm, the knowledge of the signal samples with new modulation modes can be efficiently learned in the current stage without forgetting the knowledge learned in the previous stages. The simulation results demonstrate that the proposed algorithm could achieve a recognition accuracy close to the upper bound with a much lower computing time and it outperformed the existing IL-based benchmarks.

Keywords: modulation recognition; deep learning; incremental learning; knowledge distillation

check for updates

Citation: Ma, S.; Zhang, L.; Song, Z.; Yu, W.; Liu, T. Leveraging Incremental Learning for Dynamic Modulation Recognition. *Electronics* **2024**, *13*, 3948. https://doi.org/10.3390/electronics13193948

Academic Editor: Sergio Busquets-Monge

Received: 16 September 2024
Revised: 5 October 2024
Accepted: 6 October 2024
Published: 7 October 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

Modulation recognition is the technology used to correctly identify the modulation mode of a wireless signal without any prior knowledge and is widely used in both cooperative and confrontational scenarios (see [1]). For instance, modulation recognition can be used to select a proper demodulation scheme and enhance the communication efficiency in cooperative scenarios, while in confrontational scenarios, it is extensively used in electronic warfare (see [2,3]). As wireless communication technology has advanced in recent years, the radio environment has become increasingly dynamic, imposing higher demands on the modulation-recognition capability. Generally, modulation-recognition methods can be divided into traditional methods and DL-based methods. Traditional methods recognize the modulation modes by detecting the physical or statistical characteristics of target signals (see [4,5]). These methods are structurally complex, requiring extensive expert knowledge. More importantly, they are sensitive to the radio environment, constraining their wide application, as seen in [6–8]. Recently, DL-based methods have attracted extensive attention in the field of modulation recognition and have promoted its development. DL-based

methods are typically data driven and optimize the recognition model parameters through statistical theory. DL-based methods were proven to be more robust than traditional methods in previous studies. In particular, O'Shea et al. [9] applied DL to the field of modulation recognition, demonstrating that DL methods outperform traditional methods in recognition accuracy and inference speed. Lin et al. [10] introduced the time and frequency attention mechanism into the convolution neural network (CNN) and enhanced the recognition performance with the assistance of two domain features. Ayman et al. [11] leveraged the fact that wireless signals have temporal correlations and combined a CNN and long short-term memory (LSTM) to further improve the recognition performance.

However, most of the existing DL-based modulation-recognition methods implicitly assume a relatively static scenario in which the signal samples of all the modulation modes can be completely collected in advance. In highly dynamic radio environments, new signals with different modulation modes may appear. It is clear that the emergence of new modulation modes can severely degrade the recognition performance of the well-trained model and new modulation modes cannot be correctly recognized. One straightforward way to solve this issue is to manually label the newly received wireless signal samples, fuse them into the dataset and re-train the model. Nevertheless, re-training a new model from scratch by using all the samples is time inefficient and resource consuming.

To tackle the above issue, researchers started to study the application of incremental learning (IL) to dynamic modulation recognition. In particular, IL is also called continual learning or lifelong learning and is an emerging paradigm in the field of machine learning used to accommodate new data while retaining previously acquired knowledge without re-training a new model from scratch [12]. One typical application of IL is class-increment learning, where the number of tasks cannot be completely predefined since the number of classes to be learned gradually increases over time. The main technical character of IL lies in the fact that by properly incorporating the transfer learning and knowledge distillation [13], the training of the new model that can accommodate new classes does not need to reuse all the old data and can save significant time and computing resources. These lightweight and continual learning capabilities make the IL particularly suitable for modulation recognition in dynamic radio environments in which wireless signals with new modulation modes arrive sequentially. For example, Todd M. et al. [14] applied IL to the field of modulation recognition. Specifically, the initial model therein could only recognize modulated signals with band frequency modulation and four-level pulse amplitude modulation (PAM4), and then the IL was adopted to extend the recognition capability of the model to eight-phase shift keying (8PSK), double side band (DSB) and a total of 11 modulation modes, verifying the feasibility of the IL-based modulation-recognition method. Furthermore, refs. [15,16] also studied the application of IL to the field of modulation recognition. In particular, Charles M. et al. [15] employed the bias correction (BiC) method in IL and discussed the impacts of different learning steps and storage capacities on the performance of the IL algorithm. Qi P. et al. [16] combined federated learning and IL to address the problem of heterogeneous signals that have a non-independent identical distribution across different scenarios.

We noticed that there was a key drawback in the above studies and the performance was unsatisfactory. More specially, the adopted IL algorithms typically calculated the loss of recognizing new modulation modes with the old model in the incremental training. Suppose that the sample feature distribution of the new modulation modes differs largely from that of the old modulation modes. Using the old model to calculate the loss of recognizing the new modulation modes may cause significant errors and inevitably degrades the incremental training efficiency and recognition performance. Alternatively, we designed a novel loss function by combining the distillation loss of recognizing the old modulation modes and the cross-entropy loss of recognizing new modulation modes, and propose a new IL-based modulation-recognition algorithm. The simulation results show the advantages of the proposed new IL-based modulation-recognition algorithm

over the existing ones in terms of the recognition accuracy in typical signal-to-noise ratio (SNR) regions.

The structure of this paper is as follows: Section 1 introduces the background of the modulation recognition, together with the motivation of this study; Section 2 provides the problem description; Section 3 elaborates on the design of the proposed IL-based modulation-recognition algorithm; Section 4 analyzes the performance of the proposed algorithm through extensive simulations; and Section 5 concludes the paper.

2. Problem Description

In modulation recognition, the receiver needs to accurately identify the modulation mode of a wireless signal. Suppose that there is an initial wireless signal dataset $\mathcal{D}_0$, which is pre-collected and contains S_0 modulation modes. Without loss of generality, (X_i, Y_i) is denoted as the i-th signal sample with the corresponding label (modulation mode) in D_0. Then, a modulation-recognition model $f(X_i; \Theta)$ can be constructed, where Θ is the vector of the model parameter variables. By using the dataset to train the model with a certain algorithm, such as stochastic gradient descent, the model can output the accurate modulation mode index of the input signal sample, i.e., $Y_i = f(X_i; \Theta_0^*)$, where Θ_0^* is the optimized vector of model parameter.

One main issue with the method above lies in the generalization capability, which means that the well-trained model can work for the modulation modes only in the collected dataset and may fail to recognize a wireless signal with unseen modulation modes. In practical scenarios, the radio environment is highly dynamic and new wireless signals with various modulation modes may appear from time to time, and it is quite challenging to have the access to pre-collect the signal samples with complete modulation modes in advance. Consider signal samples with $T_m (m \geq 1)$ new modulation modes that arrive and are manually labeled at the beginning of the m-th time interval. By denoting $D_{m,old}$ and $D_{m,new}$ as the datasets storing the old and new data in the beginning of the m-th time interval, the overall dataset at the beginning of the m-th time interval can be denoted as $\mathcal{D}_m = \mathcal{D}_{m,old} \cup \mathcal{D}_{m,new}$. Then, the target in the m-th time interval is to train a proper model $f(X_i; \Theta_m)$ to recognize both old and new modulation modes, i.e., $Y_i = f(X_i; \Theta_m^*)$ for $\exists$ $(X_i, Y_i) \in D_m$, where Θ_m^* is the optimized vector of the model parameters.

Using the existing IL-based modulation-recognition methods, it was shown that IL is a promising tool to achieve dynamic modulation recognition, although the current performance is still unsatisfactory. In this study, we designed a new IL-based modulation-recognition algorithm to improve the recognition performance.

3. Proposed IL-Based Modulation-Recognition Algorithm

In this part, we propose a new IL-based modulation-recognition algorithm to achieve dynamic modulation-recognition capability in the presence of continuously arriving new modulation modes. In particular, we first give the main principle of the new algorithm. Then, we elaborate on the details of the design.

3.1. Main Principle of the Proposed Method

The proposed IL-based modulation-recognition method can be divided into an initial stage and multiple incremental stages, as shown in Figure 1. In the initial stage, suppose that there are S modulation modes in the pre-collected dataset of wireless signal samples. The target of the initial stage was to establish an initial deep neural network (DNN) and train the DNN (optimize the DNN parameters) to correctly recognize the corresponding modulation modes of the input wireless signal samples. In this stage, we optimized the DNN structure in our previous work [17] and trained the DNN parameters with supervised learning, forming an initial modulation-recognition model. In each incremental stage, supposing that wireless signal samples subject to new modulation modes arrived sequentially and were manually labeled, we used the IL to update the DNN model to accommodate new modulation modes. This can be justified by the fact that multiple signal

samples with different modulation modes may arrive during a certain time interval, though only the modulation modes with enough accumulated signal samples can be used for the model update. In particular, the model update included three modules, i.e., a DNN output port argument to comply with increased modulation modes, an incremental training module for new DNN parameter optimizations and an exemplar update. It should be pointed out that the exemplar in each incremental stage is defined as the stored dataset consisting of both old and new modulation modes while dropping the other samples at the end of the current incremental stage [18], which are used as the old dataset for the training in the next incremental stage. The exemplar mechanism enables the DNN model to revisit the knowledge of partial signal samples with old modulation modes while learning the knowledge from the signal samples with new modulation modes, preventing the catastrophic forgetting issue in IL.

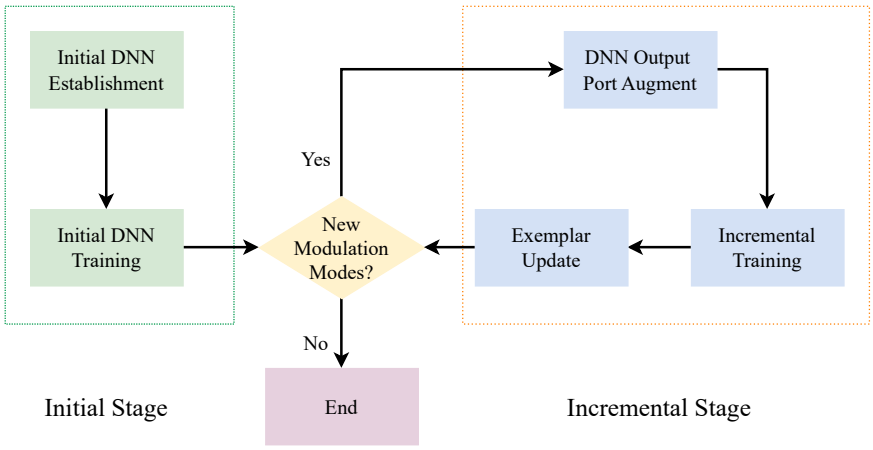


Figure 1. Proposed algorithm structure.

3.2. Initial Stage

We first needed to establish a proper DNN for efficient learning in the initial stage and the following incremental stages. Note that the designed DNN in our previous work [17] showed feature extraction capabilities superior to other existing DNN designs. In particular, as shown in Figure 2, the time-domain signal sequence in [17] is first transformed into a frequency-domain signal sequence and then both the time-domain and frequency-domain signal sequences are concurrently processed by real-value convolution and complex-value convolution, forming four data flows, which are then fused with a proper attention mechanism for adaptive multi-domain feature extractions and modulation recognition. The time-frequency attention mechanism assigns adaptive weights to different features in the time and frequency domains, allowing the algorithm to focus more on important features, thus further enhancing the recognition performance. In the proposed algorithm, the attention mechanism is influenced by the convolution and channel attention mechanisms. It first performs convolution operations on both the time-domain and frequency-domain features, then computes a global average for each channel, calculates the weights of the channel features through the Sigmoid function and finally multiplies the features by the corresponding weights to achieve efficient feature fusion.

Nevertheless, the designed DNN in [17] cannot be directly used in IL due to the imbalance between the old data and new data. Specifically, to accelerate learning the knowledge from new data in incremental learning, the new data size is typically larger than the old data size, and the total training loss of the incremental learning is the summation of the training losses of the old data and new data. Thus, there exists an inherent amount imbalance between the old and new data in the training process of incremental learning. In other words, the total training loss may be dominated by the training loss of the new data. Then, the model parameters are mainly updated in the direction of minimizing the training

loss of the new data. This causes a bias in the recognition accuracy between the old and new modulated signals [19].

To mitigate the impact of data imbalance on the performance, we added an output-correction layer after the output layer in [17] to correct the bias. To achieve this, we first removed the Softmax activation function of the neurons in the output layer of the DNN model, after which we added a new layer named the output-correction layer, as shown in Figure 2. According to [15,19], the bias can be effectively corrected by adding an output-correction layer. Without the output-correction layer, the output values subjected to the new modulation modes are much larger than those of the old modulation modes, i.e., the model tends to classify the input as new modulation modes. To solve this issue, the output-correction layer first scales the values from the output layer adaptively, and then maps the scaled values onto (0, 1). Mathematically, the processing of the output-correction layer can be described as

$$y_i = Sigmoid(w_i * x_i + b_i), \tag{1}$$

where x_i denotes the value from the i-th port in the output layer, and w_i and b_i denote the parameters of the i-th port in the output-correction layer, which are both learnable. To this end, supervised learning can be used to train the DNN model until convergence. More details can be referenced in [17].

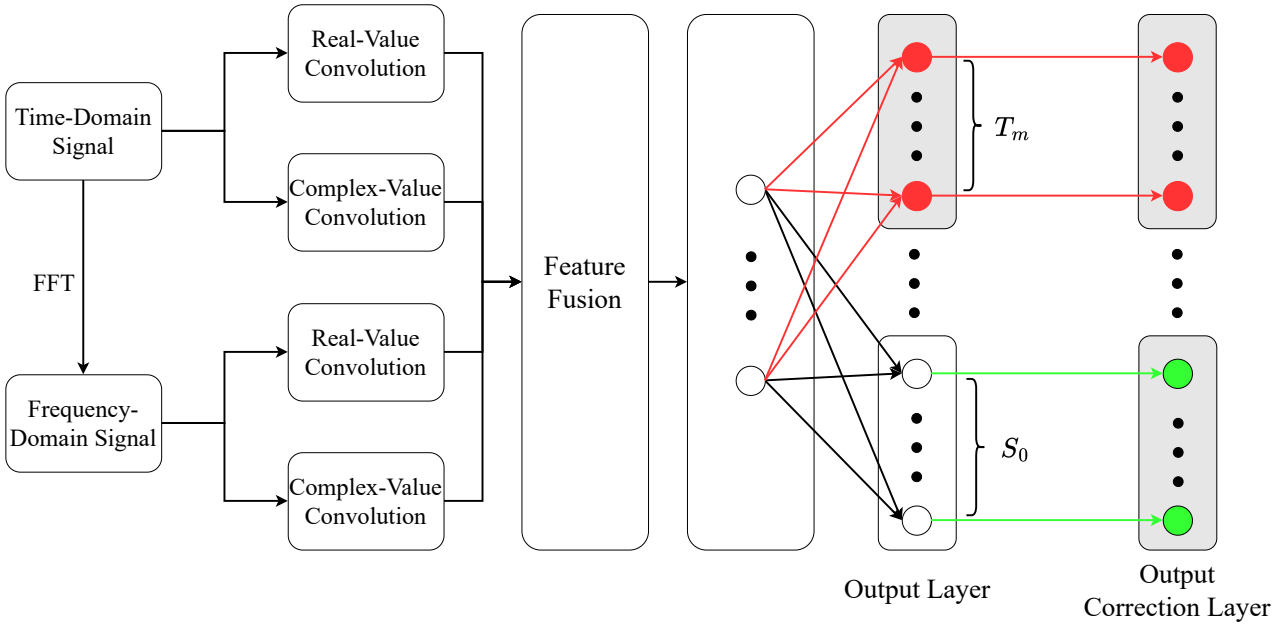


Figure 2. Designed DNN model, in which the components with a white background show the DNN structure in [17], green circles and arrows denote the newly added components in the initial stage, and the red circles and arrows denote the newly added components in the m-th incremental stage.

3.3. Incremental Stage

As mentioned above, each incremental stage includes three modules, i.e., a DNN output port argument for complying with increased modulation modes, incremental training for the parameter optimizations of the new DNN model and an exemplar update. The signal-processing procedure in each incremental stage is given in Algorithm 1. In the m-th incremental stage, the algorithm first augments the output and output correction ports based on the number of new emerging modulation modes, where the parameters are randomly initialized. Subsequently, the loss is calculated based on the input data and proposed loss function, and the parameters of the two parts of the model are alternately updated according to the corresponding gradients. Finally, the stored signals are updated based on the signal storage capacity, i.e., the exemplar is updated. In the following, we elaborate on the design of each module.

Algorithm 1: IL-based modulation-recognition algorithm in the m-th incremental stage.

Input:
$\mathcal{D}_{m,new} = (\mathcal{D}_{m,S_m+1}, \ldots, \mathcal{D}_{m,S_m+T_m})$; `// new data`
C; `// exemplar capacity`
Required:
Θ^*_{m-1}; `// well-trained parameters`
$\mathcal{D}_{m,old} = (\mathcal{D}_{m,1}, \ldots, \mathcal{D}_{m,S_m})$; `// old data in the exemplar`
DNN output port augment:
Add T_m neurons in both the output layer and the output-correction layer.
Each added neuron in the output layer is fully connected to the neurons in the previous layer and is connected uniquely to an added neuron in the output-correction layer.
Incremental training:
for $k = 1, \ldots,$ *epochs* **do**
If ($k == 1$):
Freeze $\Theta^{(1)}_{m,1}$ and fine-tune $\Theta^{(1)}_{m,2}$ with Equation (5) using
$$\Theta^{(1)}_{m,2} \leftarrow \Theta^{(1)}_{m,2} - \eta * \frac{1}{N^{(1)}_m} \sum_{(X_i,Y_i)\in\mathcal{D}^{(1)}} \nabla_{(\Theta^{(1)}_{m,2})} L(\Theta^{(1)}_{m,1}, \Theta^{(1)}_{m,2}),$$
where the loss $L(\Theta^{(1)}_{m,1}, \Theta^{(1)}_{m,2})$ is calculated with Equations (2)–(4).
elif (k is odd):
Initialization: $\Theta^{(k)}_{m,1} = \Theta^{(k-1)}_{m,1}$, $\Theta^{(k)}_{m,2} = \Theta^{(k-1)}_{m,2}$.
Freeze $\Theta^{(k)}_{m,1}$ and fine-tune $\Theta^{(k)}_{m,2}$ with Equation (5) using
$$\Theta^{(k)}_{m,2} \leftarrow \Theta^{(k)}_{m,2} - \eta * \frac{1}{N^{(k)}_m} \sum_{(X_i,Y_i)\in\mathcal{D}^{(k)}} \nabla_{(\Theta^{(k)}_{m,2})} L(\Theta^{(k)}_{m,1}, \Theta^{(k)}_{m,2}),$$
where the loss $L(\Theta^{(k)}_{m,1}, \Theta^{(k)}_{m,2})$ is calculated with Equations (2)–(4).
else:
Initialization: $\Theta^{(k)}_{m,1} = \Theta^{(k-1)}_{m,1}$, $\Theta^{(k-1)}_{m,2} = \Theta^{(k)}_{m,2}$.
Freeze $\Theta^{(k)}_{m,2}$ and train $\Theta^{(k)}_{m,1}$ with Equation (6) using
$$\Theta^{(k)}_{m,1} \leftarrow \Theta^{(k)}_{m,1} - \eta * \frac{1}{N^{(k)}_m} \sum_{(X_i,Y_i)\in\mathcal{D}^{(k)}} \nabla_{(\Theta^{(k)}_{m,1})} L(\Theta^{(k)}_{m,1}, \Theta^{(k)}_{m,2}),$$
where the loss $L(\Theta^{(k)}_{m,1}, \Theta^{(k)}_{m,2})$ is calculated with Equations (2)–(4).
end
Exemplar update:
$C_m \leftarrow K/(S_m + T_m)$
Randomly select C_m samples of each modulation mode from $\mathcal{D}_m$ and store them in the exemplar $\mathcal{D}_{m+1,old}$.

3.3.1. Dnn Output Port Argument

It is clear that if we wanted to use the well-trained DNN in the previous stage to recognize the wireless signal samples with new modulation modes, we needed to revise the old DNN structure. To achieve this, we denoted the number of modulation modes in the beginning of the old data (exemplar) and new data in the m-th stage as S_m and T_m, respectively, and we added T_m neurons without adopting any activation function in the output layer, as shown in Figure 2. Accordingly, we also added T_m neurons with Sigmoid activation functions in the output-correction layer. Then, there are $S_m + T_m$ neurons in both the output layer and the output-correction layer, which corresponds to $S_m + T_m$ modulation modes (including S_m old modulation modes and T_m new modulation modes). In particular, each added neuron in the output layer is fully connected to the neurons in the previous layer and is connected uniquely to an added neuron in the output-correction layer.

3.3.2. Incremental Training

The incremental training module aims to optimize the parameters of the revised DNN model (new DNN model) after the output port arguments with both old data (exemplar) and new data. The main idea of the incremental training is twofold. First, since a part of the revised DNN model in the current stage is the same as the old DNN model in the previous stage, the knowledge represented by the old DNN model in the previous stage can be transferred to the new DNN in the current stage, i.e., the well-trained parameters in the old DNN model can be reused in the new DNN model with minor updates. This operation helps to accelerate the training of the new DNN model with partial old data. Second, both the exemplar and new data are jointly utilized to optimize the revised DNN model so that the knowledge in new modulation modes can be learned without forgetting the knowledge in old modulation modes. This can prevent the catastrophic forgetting issue.

Here, we denote $\Theta_m = [\Theta_{m,1}, \Theta_{m,2}]$, in which $\Theta_{m,1}$ represents the vector of the parameter variables before the DNN output port arguments and $\Theta_{m,2}$ represents the introduced vector of the parameter variables due to the DNN output port argument in the m-th stage. Since the vector of the parameter variables before the DNN output port argument in the m-th stage is identical to the vector of the parameter variables after the DNN output port argument in the $(m-1)$-th stage, we have $\Theta_{m,1} = \Theta_{m-1}$. To train Θ_m, the two sub-vectors in Θ_m, i.e., $\Theta_{m,1}$ and $\Theta_{m,2}$, are alternatively trained until both of them converge in the m-th stage.

To begin with, an epoch is defined as the duration of training $\Theta_{m,1}$ or $\Theta_{m,2}$ once in an incremental training stage, the epoch index is defined as k, and the initial values of $\Theta_{m,1}$ and $\Theta_{m,2}$ in the k-th epoch of the m-th stage are defined as $\Theta_{m,1}^{(k)}$ and $\Theta_{m,2}^{(k)}$. It is clear that multiple epochs are needed for both $\Theta_{m,1}$ and $\Theta_{m,2}$ to converge in the m-th stage and the converged values of $\Theta_{m,1}$ and $\Theta_{m,2}$ in the m-th stage are denoted as $\Theta_{m,1}^*$ and $\Theta_{m,2}^*$.

In the first epoch (i.e., $k = 1$) of the m-th stage, the knowledge from the $(m-1)$-th stage is transferred to the m-th stage by initializing $\Theta_{m,1}$ as $\Theta_{m,1}^{(1)} = \Theta_{m-1}^*$. Meanwhile, $\Theta_{m,2}^{(1)}$ is randomly initialized. Then, a mini-batch $D_m^{(k)} = D_{m,old}^{(k)} \cup D_{m,new}^{(k)}$ with the size $N_m^{(k)}$ is randomly sampled from the datasets D_m, where $D_{m,old}^{(k)}$ and $D_{m,new}^{(k)}$ denote the samples from the dataset $D_{m,old}$ and $D_{m,new}$ in the k-th epoch of the m-th stage, respectively. After this, we designed a novel loss function as

$$L(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}) = L_D^{old}(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}) + L_C^{new}(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}), \tag{2}$$

where the first term on the right is the distillation loss of recognizing old modulation modes and is defined by

$$\begin{aligned} &L_D^{old}(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}) \\ &= -\sum_{(X_i,Y_i)\in\mathcal{D}_{m,old}^{(k)}} \Bigg[\sum_{y=S_m+1}^{y=S_m+T_m} \log\Big(1 - f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big)\Big) + \\ &\sum_{y=1}^{S_m} \Big(f_{m-1,y}^*\big(X_i; \Theta_{m-1,1}^*, \Theta_{m-1,2}^*\big) \log f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big) + \\ &\Big(1 - f_{m-1,y}^*\big(X_i; \Theta_{m-1,1}^*, \Theta_{m-1,2}^*\big)\Big) \log\Big(1 - f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big)\Big)\Big)\Bigg], \end{aligned} \tag{3}$$

and the second term of Equation (2) on the right denotes the cross-entropy loss of recognizing the new modulation modes and is defined by

$$
\begin{aligned}
&L_C^{new}(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}) \\
&= -\sum_{(X_i,Y_i)\in\mathcal{D}_{m,new}^{(k)}} \Bigg[\sum_{y=1}^{y=S_m+T_m} \Bigg(\delta_{y,Y_i} \log f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big) + \\
&\big(1-\delta_{y,Y_i}\big) \log f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big) \Bigg) \Bigg],
\end{aligned} \tag{4}
$$

where X_i denotes the i-th modulated signal in the dataset, while Y_i is the corresponding modulation mode index. δ_{y,Y_i} is one for $Y_i = y$ and is zero otherwise. $f_{m,y}^{(k)}(\cdot)$ denotes the value at the y-th port of the output-correction layer in the k-th epoch of the m-th stage, while $f_{m-1,y}^{*}(\cdot)$ denotes the value at the y-th port of the output-correction layer in the well-trained model in the $(m-1)$-th stage.

Then, $\Theta_{m,1}^{(k)}$ is frozen and $\Theta_{m,2}^{(k)}$ is updated in the direction of minimizing the loss function, i.e.,

$$
\Theta_{m,2}^{(k)} \leftarrow \Theta_{m,2}^{(k)} - \eta * \frac{1}{N_m^{(k)}} \sum_{(X_i,Y_i)\in\mathcal{D}^{(k)}} \nabla_{(\Theta_{m,2}^{(k)})} L(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}), \tag{5}
$$

where η denotes the learning rate and $\nabla_{(\Theta_{m,2}^{(k)})} L(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)})$ denotes the partial derivative of the loss function with respect to $\Theta_{m,2}^{(k)}$.

In the second epoch ($k = 2$), $\Theta_{m,1}^{(2)} = \Theta_{m,1}^{(1)}$ and $\Theta_{m,1}^{(2)} = \Theta_{m,1}^{(1)}$ are initialized, and a mini-batch $D_m^{(k)} = D_{m,old}^{(k)} \cup D_{m,new}^{(k)}$ with size $N_m^{(k)}$ is randomly sampled and the loss is calculated with Equations (2)–(4). To this end, $\Theta_{m,1}^{(k)}$ is updated in the direction of minimizing the loss, i.e.,

$$
\Theta_{m,1}^{(k)} \leftarrow \Theta_{m,1}^{(k)} - \eta * \frac{1}{N_m^{(k)}} \sum_{(X_i,Y_i)\in\mathcal{D}^{(k)}} \nabla_{(\Theta_{m,1}^{(k)})} L(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}). \tag{6}
$$

For $k > 2$, Equations (5) and (6) are used to update $\Theta_{m,2}^{(k)}$ and $\Theta_{m,1}^{(k)}$ when k is odd and even, respectively. This incremental training process continues until both $\Theta_{m,1}$ and $\Theta_{m,2}$ converge to $\Theta_{m,1}^{*}$ and $\Theta_{m,2}^{*}$, respectively, which leads to $\Theta_m^{*} = [\Theta_{m,1}^{*}, \Theta_{m,2}^{*}]$.

It should be noted that the proposed IL-based modulation-recognition algorithm with the newly designed loss function from Equations (2)–(4) is different from the existing IL-based modulation-recognition algorithm with the following distillation loss, i.e.,

$$
\begin{aligned}
&L(\Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}) \\
&= -\sum_{(X_i,Y_i)\in\mathcal{D}_m^{(k)}} \Bigg[\sum_{y=S+1}^{S+T} \Bigg(\delta_{y,Y_i} \log f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big) \\
&+ \big(1-\delta_{y,Y_i}\big) \log\Big(1 - f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big)\Big) \Bigg) \\
&+ \sum_{y=1}^{S} \Bigg(f_{m-1,y}^{*}\big(X_i; \Theta_{m-1,1}^{*}, \Theta_{m-1,2}^{*}\big) \log f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big) \\
&+ \Big(1 - f_{m-1,y}^{*}\big(X_i; \Theta_{m-1,1}^{*}, \Theta_{m-1,2}^{*}\big)\Big) \log\Big(1 - f_{m,y}^{(k)}\Big(X_i; \Theta_{m,1}^{(k)}, \Theta_{m,2}^{(k)}\Big)\Big) \Bigg) \Bigg].
\end{aligned} \tag{7}
$$

By comparing Equation (7) to Equations (2)–(4), it is clear that Equation (7) ignores the fact that the well-trained DNN model in the previous stage (old DNN model) contains little knowledge of the new arriving modulation modes. Using the old DNN model to calculate

the loss of recognizing the new modulation may cause significant errors and degrade the training efficiency. In contrast, our designed loss function evaluates the loss components of recognizing the old and new modulation modes separately and is able to reveal a more correct loss, which enhances the incremental training efficiency.

3.3.3. Exemplar Update

To prevent the catastrophic forgetting issue [20] of IL, i.e., the DNN model may loss the learned knowledge when learning new knowledge, the exemplar mechanism is widely used in the classical IL algorithm. In other words, after each incremental training stage, only partial sample data are stored in the memory as old data for the training of the next incremental stage. The selection criteria of the exemplars is twofold. First, both the amount and quality of the wireless signal samples subject to different modulation modes should be balanced to guarantee the performance of each incremental learning stage. Second, the memory is typically limited and the total amount of stored data should not exceed a specific threshold.

Although the number of stored signal samples can be easily guaranteed, the quality of wireless signal samples for IL is hard to correctly define. There exist two widely used exemplar-selection methods based on different understandings of the quality of wireless signal samples for IL, i.e., the random-selection algorithm and herding algorithm [18]. In particular, the random-selection algorithm randomly picks a fixed number of sample data of each modulation mode to store and drops the other data. This method implicitly assumes that the data diversity has a larger impact on the performance. The herding algorithm [18] picks the samples of a specific modulation mode according to its feature similarity to the center of all the sample features with the same modulation modes. This method emphasizes that the data feature is more important to the IL performance.

In this study, we considered three candidates for the exemplar selection, i.e., the random-selection algorithm, the herding algorithm and the SNR-based algorithm. In particular, the SNR-based exemplar-selection algorithm stores the samples with the highest SNR. Then, we identified the random exemplar-selection algorithm as the optimal one for modulation recognition through simulations in the next section, i.e., the sample dataset $\mathcal{D}_{m+1,old}$ in the beginning of the $(m+1)$-th stage was constructed by randomly storing a fixed number of each modulation mode in $\mathcal{D}_m$ at the end of the m-th stage.

4. Results Discussion

In this part, we provide simulation results to verify the advantages of the proposed algorithm. In particular, we first introduce the benchmarks and then give the simulation setup, followed by the performance comparison and analysis with benchmarks.

4.1. Benchmarks

In the field of modulation recognition, the term "automatic" is used to describe a modulation-recognition model that has appropriately learned the features of different modulated signals in advance and can recognize different modulated signals without the help of a human. The term "dynamic" is typically used to describe the radio environment, in which unseen modulated signals may appear. Although dynamic modulation recognition generally belongs to the automatic modulation-recognition category while considering a special scenario in which modulated signals arrive sequentially, conventional automatic modulation-recognition algorithms cannot be directly used for fair comparisons. Instead, we adopted two widely used upper-bound and lower-bound (i.e., all-data learning and no-data learning) IL-based algorithms to verify the advantages of the proposed algorithm. Meanwhile, two existing IL-based modulation-recognition algorithms (i.e., iCaRL and BiC) were also used as benchmarks to demonstrate the superiority of the proposed algorithm.

- **All-data learning:** The all-data learning method, which is also known as the joint-learning method, assumes infinite memory capacity and stores all historical data for IL, allowing for using all old and new data during training, where new data are

considered the old data in the next incremental stage. It is clear that this method assumes an ideal condition and the corresponding performance is typically used as the upper bound of other methods (see [12,15]).

- **No-data learning:** In contrast to all-data learning, no-data learning considers the worst-case scenario in which there is no memory to store any old data and only new data can be used for the incremental training. It should be noted that the no-data learning shares the same idea as sequential learning, which learns different modulation modes in different stages and only new data are utilized in each stage. It is clear that this method suffers from a severe catastrophic forgetting issue since each incremental training stage ignores the knowledge of the old data. Then, the performance of this method can be used as the lower bound of other methods.
- **iCaRL:** The iCaRL algorithm proposes to prevent the catastrophic forgetting issue by using the herding algorithm. In addition, the algorithm also adopts knowledge distillation technology to enable the new DNN model to learn from both old and new modulation modes. More details can be referenced in [18].
- **BiC:** The BiC algorithm adopts random exemplar selection to prevent the catastrophic forgetting issue. In addition, the algorithm proposes to add an output-correction layer to deal with the data imbalance between the old and new data. More details about this algorithm can be found in [19].

4.2. Simulation Setup

It should be pointed out that we considered both typical communication modulation modes and radar modulation modes in the experiments, meaning that our proposed algorithm can be used in both communication and radar scenarios, even in integrated communication and radar scenarios. Therefore, this study considered 13 modulation modes, which were binary phase shift key (BPSK), quadrature PSK (QPSK), 8PSK, continuous phase frequency shift key (CPFSK), Gaussian FSK (GFSK), DSB, single side band (SSB), linear frequency modulation (LFM), 16 quadrature amplitude modulation (16QAM), 64QAM, pulse compression modulation (PCM), wideband frequency modulation (WBFM) and PAM4 in the order of learning. The simulated channel environment was a Rice channel with a K-factor of four, and eight sinusoids were used to simulate the scattering path. The SNR ranged from -5 dB to 15 dB, with 2000 frames per mode at each SNR and 128 sampling points per frame. In addition, the bandwidth of the signal was 100 MHz, the carrier frequency was 2.4 GHz and the sampling frequency was 400 MHz. Considering the relative movement between the signal emitter and the signal receiver, the maximum Doppler shift was set to 20 KHz.

Here, the ratio of the dataset for training and validation was 7:3, and the simulation was implemented by using an Nvidia RTX 3080Ti GPU manufactured by Micro-Star in Chengdu, China. based on Pytorch 2.0.1. During the training, the batch size was set to 256 and the adopted optimizer was adaptive moment estimation (Adam) with a learning rate of 0.001. The exponential decay rates for the first and second moment estimates, i.e., $\beta 1$ and $\beta 2$, were 0.9 and 0.999, respectively. In the initial stage and each incremental stage, the parameters Θ_1 and Θ_2 were trained for five epochs each. The number of modulation modes in the initial stage was three, and two new modulation modes appeared at each incremental stage. The modulation modes in the initial stage and new modulation modes in each incremental stage are listed in Table 1. Without special instructions, the random exemplar-selection method was adopted and the total memory capacity was 29,400 samples. The main parameters in the simulation are summarized in Table 2.

In real-world applications, datasets may be much larger and contain more modulation modes. It is worth noting that our proposed algorithm can be applied to a dataset with any size and modulation mode. In other words, our proposed algorithm can be directly used in real-world applications. In fact, a larger dataset may enhance the training quality in the field of DL.

Table 1. New modulation modes in different stages.

Stage	New Modulation Modes
Initial stage	BPSK, QPSK, 8PSK
Incremental stage 1	CPFSK, GFSK
Incremental stage 2	SSB, DSB
Incremental stage 3	LFM, 16QAM
Incremental stage 4	64QAM, PCM
Incremental stage 5	WBFM, PAM4

Table 2. Parameter settings.

Parameter Name	Value
Bandwidth	100 MHz
Carry frequency	2.4 GHz
Sample frequency	400 MHz
Doppler shift	20 KHz
K-factor	4
SNRS	−5:1:15 dB
Samples of each modulation mode	2000 × 21
Training and validation ratio	7:3
Epochs	10
Exemplar capacity	29,400
Batch size	256
Optimizer	Adam
Learning rate	0.001
$\beta 1$	0.9
$\beta 2$	0.999

4.3. Performance Comparison and Analysis

Figure 3 shows the recognition performance curves of the different algorithms, where the all-data and no-data learning methods were used as the upper and lower bounds, respectively. It can be seen that the recognition performance curves of the proposed algorithm and the all-data method (the upper bound) were close, had the same trend and were higher than the other benchmarks. In particular, as the number of incremental modulation modes grew, the average recognition accuracy first increased and then decreased. This was because if the new modulation modes were easier to recognize than the old modulation modes, the recognition accuracy of the new modulation modes was higher than that of the old modulation modes, leading to a higher average recognition accuracy. Otherwise, the recognition accuracy of the new modulation modes was lower than that of the old modulation modes, leading to a lower average recognition accuracy. To be specific, the recognition accuracy gap between the proposed algorithm and the all-data (upper-bound) algorithm was no higher than 5% in the considered setup. Meanwhile, the proposed algorithm outperformed the iCaRL by up to about 9% and outperformed the BiC algorithm by up to about 7%. For fair comparisons, the performances of the different algorithms were evaluated with the same method in the same hardware and software environment and the results also coincide with our analysis.

The all-data method stores all the historical data, which naturally results in a good performance. However, this method comes with the issues of increased memory requirements and a longer training time. The average training time of the all-data method and the proposed algorithm are shown in Table 3. It can be observed that the average training time of the proposed algorithm was much shorter than that of the all-data method as the number of new modulation modes grew. Furthermore, the average training time in each incremental stage almost remained constant and was higher than that in the initial stage. This was because the knowledge distillation in each incremental stage caused additional computing complexity compared with the training in the initial stage.

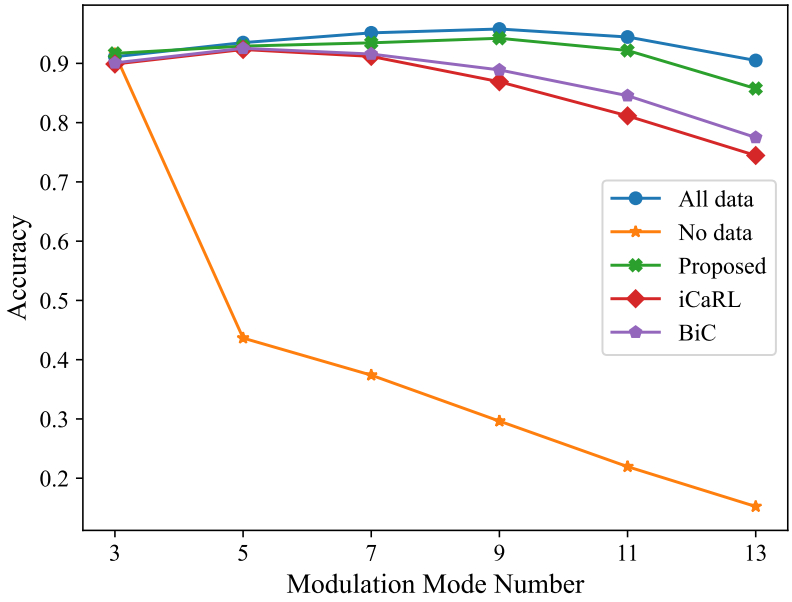


Figure 3. Performances of different algorithms.

In addition, compared with the conventional methods, such as the all-data algorithm, the proposed algorithm could relieve the burden on the computational resource, which was typically reflected in the training time, as shown in Table 3. Indeed, the computational resource constraints could be further tackled by properly reducing the model complexity with ignorable recognition accuracy loss. For real-time application scenarios, the inference time of the model is the main metric and can be largely reduced with parallel computing, such as in a field-programmable gate array or an application-specific integrated circuit [21].

Table 3. Average training time of each epoch (s/epoch), where "No. of mod." denotes number of modulation modes.

Algorithm \ No. of mod.	3	5	7	9	11	13
Proposed	4.09	6.58	6.80	6.81	6.69	6.77
All data	4.10	11.1	15.96	19.89	25.18	29.09

4.4. Recognition Performances with Different Loss Functions

As mentioned above, the proposed IL-based modulation-recognition algorithm adopted a newly designed loss function. Then, we compared the performance of the IL-based modulation-recognition algorithm with different loss functions in Figure 4. It can be observed that the performance of the IL-based modulation-recognition algorithm with the newly designed loss function was generally better than the algorithms with the no knowledge distillation and conventional knowledge distillation loss functions. This result demonstrates the advantages of the proposed loss function. In particular, the performance gap became large as the number of new modulation modes grew. This was because as the number of modulation modes grew, the recognition task became more complex and it was more challenging for the existing algorithms to distinguish different modulation modes correctly.

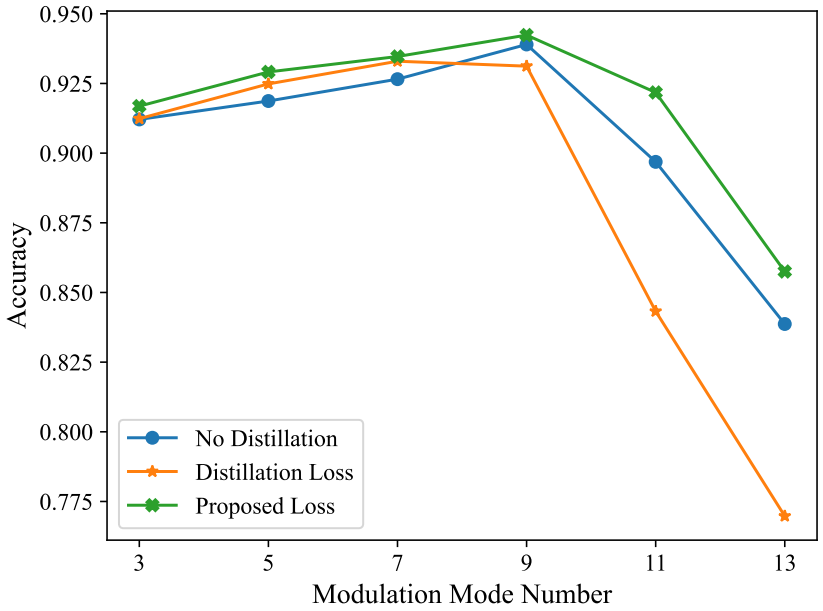


Figure 4. Performances with different loss functions.

4.5. Recognition Performances with Different Exemplar-Selection Schemes

To determine a proper exemplar-selection scheme for the proposed IL-based modulation-recognition algorithm, we provide the performance of the proposed algorithm with different exemplar-selection schemes, including the random-selection scheme, herding algorithm and the SNR-based scheme, in Figure 5. In the figure, the performance of the random-selection strategy is higher than the other two schemes. In particular, the herding algorithm emphasizes the central features instead of diversities. In practical situations of modulation recognition, the highly dynamic radio environment leads to non-central features in signal samples and the diversity is more important than the central features for modulation-recognition tasks. Similar statements apply for the phenomenon where the random-selection scheme was better than the SNR-based scheme, which can be verified by Figure 6, in which the SNR-based scheme worked well only in the high-SNR region.

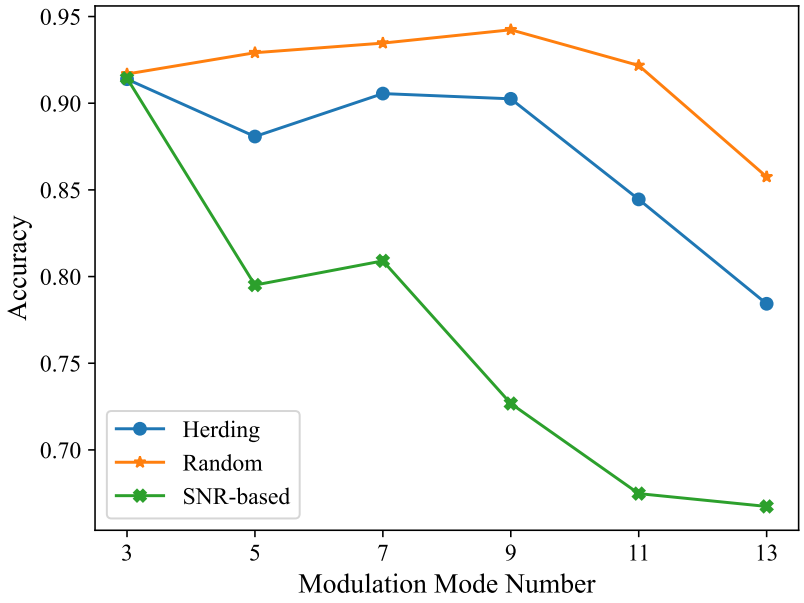


Figure 5. Performances with different selection schemes.

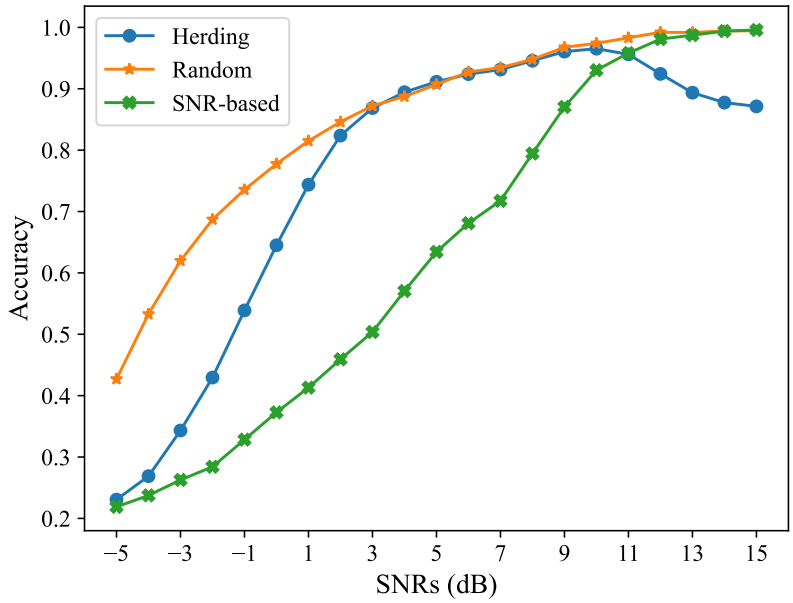


Figure 6. Performances with different selection schemes and SNRs.

4.6. Recognition Performances with Different Memory Capacities

Using memory to store exemplars can effectively prevent the catastrophic forgetting issue, and the trade-off between the memory size and accuracy is also worth studying. Figure 7 shows the impacts of different memory sizes on the performance, where the memory size to store exemplars was set to 2, 1, 0.5, 0.3 and 0.1 times the 29,400 samples. Additionally, the all-data and no-data methods were still used as the upper and lower bounds. It can be seen that as the size of the exemplar increased, the performance of the IL-based modulation-recognition algorithm improved, demonstrating that storing more historical samples could effectively deal with the catastrophic forgetting issue. It is worth noting that even with only 2940 (0.1 times) samples in the exemplar, the accuracy still exceeded 70% when recognizing 13 modulation modes, which was significantly higher than the accuracy of the no-data method.

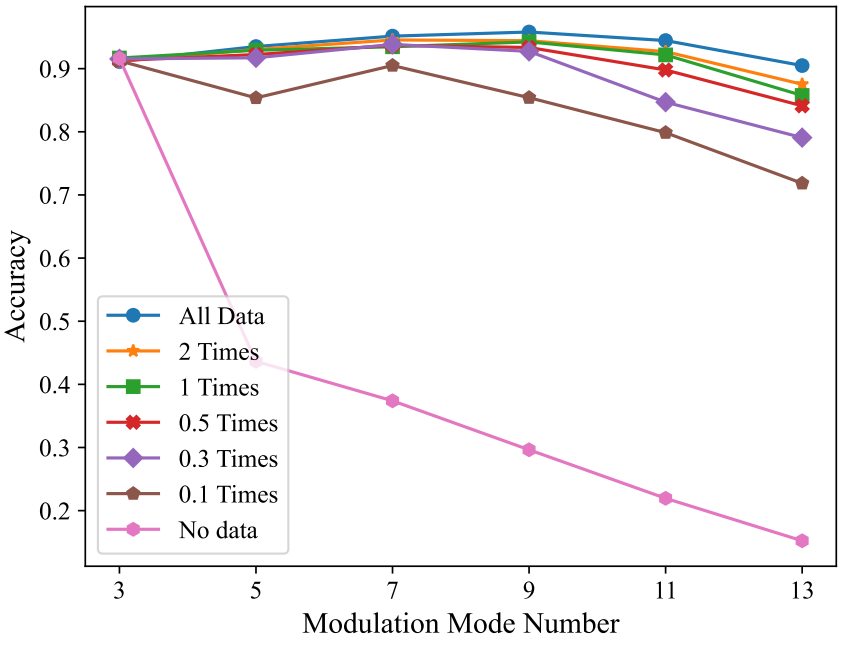


Figure 7. Performances with different memory sizes.

5. Conclusions

This study investigated the dynamic modulation recognition in a practical situation in which the signal samples with new modulation modes could appear sequentially and it was demanding to completely collect the signal sample of all the modulation modes in advance. To tackle this issue, we proposed a novel IL-based modulation-recognition method, which consists of an initial stage and multiple incremental stages. The main novelty of the proposed method lies in the new loss function design in each incremental stage, which combines the distillation loss of recognizing old modulation modes and the entropy loss of recognizing new modulation modes. With the proposed algorithm, the knowledge of the signal samples with new modulation modes could be efficiently learned while retaining the knowledge learned in the previous stage. The simulation results demonstrated that the proposed algorithm could achieve a recognition accuracy close to the upper bound with a much lower computing time and it outperformed the existing IL-based benchmarks.

It is worth pointing out that thanks to the recent advances in graph processing, the graph-based modulation-recognition algorithm that utilizes the spatial–spectral features of modulated signals is a promising research direction. Nevertheless, this kind of algorithm typically suffers from longer training and inference times. In our future work, we will leverage the advances in the graph-processing field and develop a lightweight graph-based modulation-recognition algorithm.

Author Contributions: Writing—original draft, S.M.; writing—review and editing, L.Z.; investigation, Z.S.; methodology, W.Y.; conceptualization, T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Basic Research and Innovation Fund in national key laboratory of wireless communications under grant IFN20230102.

Data Availability Statement: Data available upon request from the authors.

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

References

1. Peng, S.; Sun, S.; Yao, Y. A survey of modulation classification using deep learning: Signal representation and data preprocessing. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 7020–7038. [CrossRef] [PubMed]
2. Wang, Y.; Gui, G.; Ohtsuki, T.; Adachi, F. Multi-task learning for generalized automatic modulation classification under non-gaussian noise with varying SNR conditions. *IEEE Trans. Wireless Commun.* **2021**, *20*, 3587–3596. [CrossRef]
3. Al-Hraishawi, H.; Chougrani, H.; Kisseleff, S.; Lagunas, E.; Chatzinotas, S. A survey on nongeostationary satellite systems: The communication perspective. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 101–132. [CrossRef]
4. Xu, J.; Sun, X.; Liu, L.; Li, Y.; Wang, Y.; Zhang, T. Modulation recognition based on instantaneous feature fusion. In Proceedings of the 2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS), Chengdu, China, 7–9 July 2023; pp. 632–638.
5. Wang, Y.; Liu, Z.; Xu, Y. Application of frequency-domain features and high-order cumulants in ANN-based communication modulation recognition. In Proceedings of the 2022 IEEE 4th International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Dali, China, 12–14 October 2022; pp. 1288–1293.
6. Zhang, Z.; Luo, H.; Wang, C.; Gan, C.; Xiang, Y. Automatic modulation classification using CNN-LSTM based dual-stream structure. *IEEE Trans. Veh. Technol.* **2020**, *69*, 13521–13531. [CrossRef]
7. Zheng, S.; Qi, P.; Chen, S.; Yang, X. Fusion methods for CNN-based automatic modulation classification. *IEEE Access* **2019**, *7*, 66496–66504. [CrossRef]
8. Hou, C.; Liu, G.; Tian, Q.; Zhou, Z.; Hua, L.; Lin, Y. Multisignal modulation classification using sliding window detection and complex convolutional network in frequency domain. *IEEE Internet Things J.* **2022**, *9*, 19438–19449. [CrossRef]
9. O'Shea, T.J.; Corgan, J.; Clancy, T.C. Convolutional radio modulation recognition networks. In Proceedings of the 17th International Conference Engineering Applications of Neural Networks, Aberdeen, UK, 2–5 September 2016; pp. 213–226.
10. Lin, S.; Zeng, Y.; Gong, Y. Learning of time-frequency attention mechanism for automatic modulation recognition. *IEEE Wireless Commun. Lett.* **2022**, *11*, 707–711. [CrossRef]
11. Emam, A.; Shalaby, M.A.; Bakr, H.E.A.; Mansour, H.A.A. A comparative study between CNN, LSTM, and CLDNN models in the vontext of radio modulation classification. In Proceedings of the 2020 12th International Conference on Electrical Engineering (ICEENG), Cairo, Egypt, 7–9 July 2020; pp. 190–195.

12. Owfi, A.; Abbasi, A.; Afghah, F.; Ashdown, J.; Turck, K. Dynamic online modulation recognition using incremental learning. In Proceedings of the 2024 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 19–22 February 2024; pp. 412–416.
13. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *Comput. Sci.* **2015**, *14*, 38–39.
14. 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.
15. Montes, C.; Morehouse, T.; Zhou, R. Class-incremental learning for baseband modulation classification: A comparison. In Proceedings of the 2024 International Wireless Communications and Mobile Computing (IWCMC), Ayia Napa, Cyprus, 27–31 May 2024; pp. 1370–1375.
16. Qi, P.; Zhou, X.; Ding, Y.; Zheng, S.; Jiang, T.; Li, Z. Collaborative and incremental learning for modulation classification with heterogeneous local dataset in cognitive IoT. *IEEE Trans. Green Commun. Netw.* **2023**, *7*, 881–893. [CrossRef]
17. Li, T.; Liu, T.; Song, Z.; Zhang, L.; Ma, Y. Deep learning-based multi-feature fusion for communication and radar signal sensing. *Electronics* **2024**, *13*, 1872. [CrossRef]
18. Rebuffi, S.A.; Kolesnikov, A.; Sperl, G.; Lampert, C.H. iCaRL: Incremental classifier and representation learning. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5533–5542.
19. 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 (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 374–382.
20. 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.
21. Woo, J.; Jung, K.; Mukhopadhyay, S. Efficient hardware design of DNN for RF signal modulation recognition. In Proceedings of the 2024 IEEE International Symposium on Circuits and Systems (ISCAS), Singapore, 19–22 May 2024; pp. 1–5.

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.