Class-Incremental Learning for Wireless Device Identification in IoT

Yongxin Liu, Jian Wang, Jianqiang Li, Shuteng Niu, and Houbing Song, *Senior Member, IEEE*

***Abstract*—Deep Learning (DL) has been utilized pervasively in the Internet of Things (IoT). One typical application of DL in IoT is device identification from wireless signals, namely Non-cryptographic Device Identification (NDI). However, learning components in NDI systems have to evolve to adapt to operational variations, such a paradigm is termed as Incremental Learning (IL). Various IL algorithms have been proposed and many of them require dedicated space to store the increasing amount of historical data, and therefore, they are not suitable for IoT or mobile applications. However, conventional IL schemes can not provide satisfying performance when historical data are not available. In this paper, we address the IL problem in NDI from a new perspective, firstly, we provide a new metric to measure the degree of topological maturity of DNN models from the degree of conflict of class-specific fingerprints. We discover that an important cause for performance degradation in IL enabled NDI is owing to the conflict of devices' fingerprints. Second, we also show that the conventional IL schemes can lead to low topological maturity of DNN models in NDI systems. Thirdly, we propose a new Channel Separation Enabled Incremental Learning (CSIL) scheme without using historical data, in which our strategy can automatically separate devices' fingerprints in different learning stages and avoid potential conflict. Finally, We evaluated the effectiveness of the proposed framework using real data from ADS-B (Automatic Dependent Surveillance-Broadcast), an application of IoT in aviation. The proposed framework has the potential to be applied to accurate identification of IoT devices in a variety of IoT applications and services. Data and code available at IEEE Dataport (DOI: 10.21227/1bxc-ke87) and https://github.com/pcwhy/CSIL.**

***Index Terms*—Internet of Things, Cybersecurity, Big Data Analytics, Non-cryptographic identification, Zero-bias Neural Network, Deep Learning.**

I. Introduction

The Internet of Things (IoT) is characterized by the interconnection and interaction of smart objects (objects or devices with embedded sensors, onboard data processing capabilities, and means of communication) to provide applications and services that would otherwise not be possible [1]–[3]. The convergence of sensors, actuators, information, and communication technologies in IoT produces massive amounts of data that need to be sifted through to facilitate reasonably accurate decision-making and control [4]–[7]. A typical way to implement smart decision functionality in IoT is by integrating learning-enabled components through Deep Learning (DL) and Deep Neural Networks (DNNs). One typical application of DNNs in IoT is the passive identification of IoT devices through their wireless signals for Non-cryptographic Device Identification (NDI) and Physical Layer authentication [8]–[10]. DL and DNNs are effective in wireless device identification under various scenarios, however, DNN models in these applications need to continuous evolving to adapt to operational variations as new devices (as new classes) are emerging. Such a continuous evolving scheme is termed as Lifelong or Incremental Learning (IL). Conventional approaches require periodic retraining to update DNNs. In this paradigm, DNNs are initialized from scratch and trained with all past and present devices' signals. Even though the best accuracies are guaranteed in these Non-Incremental Learning (Non-IL) schemes, the memory consumption and training time can grow drastically as new devices are added in. Therefore, there is a need for IL with a reasonable balance between accuracy, memory consumption, and training efficiency. In IoT, less or zero memory for historical data are preferred during the continuous evolving [11].

Compared to conventional non-incremental learning (non-IL) schemes, DNN models can only use a very small proportion or even none of the data from the previous stages, a.k.a. old tasks, while they are trained to recognize new devices. The absence of data from old tasks results in *Catastrophic Forgetting*, a phenomenon of significant degradation of accuracy after training on new tasks. IL has become an emerging topic in machine learning, however, many of the methods are not adaptable in IoT. For example, some works require storing specifically chosen old data, and can consume a large amount of memory [12] gradually. Other works require incrementally training task-related generative models for knowledge replay, but these generative models require notorious efforts [13]. In addition, there are several attempts to either use regularization or knowledge distillation to implement memoryless methods to prevent DNNs from forgetting [11]. Balancing between learning and forgetting is difficult, especially when the internal mechanism of catastrophic forgetting is not yet clear. Besides, there is a lack of theoretic explanation to explore the difference between the key characteristics between IL and regularly trained models.

In this paper, we explored the topological properties of fingerprints in the final classification layers of DNN-enabled wireless device identification models after IL and regular training, we discovered that the main cause of catastrophic forgetting is due to the nonoptimal distribution of feature vectors and their reprentatives (fingerprints) in the latent space. Based on the discoveries, we designed an enhanced IL scheme,

Yongxin Liu, Jian Wang, Shuteng Niu, and Houbing Song are with the Security and Optimization for Networked Globe Laboratory (SONG Lab), Embry-Riddle Aeronautical University, Daytona Beach, FL 32114 USA

Jianqiang Li is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, Guangdong 518060 China

Corresponding author: Houbing Song

Manuscript received March 22, 2021; revised XXX.

the Channel Separation Enabled Incremental Learning (CSIL), for wireless device identification systems. We manually introduced separations in representative spaces between different tasks (learning stages). The effectiveness of the proposed framework in massive signal recognition and improving the incremental learning performance has been demonstrated. The contributions of this paper are as follows:

- We provide a new metric, the Degree of Conflict (DoC), to quantitatively analyze the topological maturity of DNN models. Using this metric, we discover that DNN models trained by conventional IL mechanisms are with low topological maturity. This metric is helpful in understanding the internal mechanisms of DNNs.
- We provide a new perspective for the causality, the conflict of fingerprints, to explain the catastrophic forgetting in DNN models.
- We provide an enhanced IL strategy, CSIL for incremental learning for DNN-enabled IoT device identification systems and test the CSIL mechanism using real signal datasets.

Our research offers not only a solution for accurate identification of IoT devices, but also useful for future development of IL for DNNs. To our best knowledge, this is the first study that jointly explores DNN and IL in Signal Intelligence Applications. Right before the publication of this work, we realized that our algorithm actually has solid evidence from the most recent advancement of neural science [14]. We share some similar findings as in [14], but from a totally different perspective and a non-biological road map. In addition, we provide the mathematical proof and are delighted to find an elegant connection between biological and artificial intelligence.

The remainder of this paper is organized as follows: A literature review of wireless device identification and incremental learning is presented in Section II. We formulate our problem in Section III with the methodology presented in Section III-D. Performance evaluation is presented in Section IV with conclusions in Section V.

II. Related works

In this section, we will provide a brief review of wireless device identification in IoT and Incremental Learning in deep learning.

A. Wireless device identification in IoT

Specific device identification is emerging as a solution to Physical layer security of IoT. The methods aim to recognize IoT devices based solely on their signals. Corresponding methods can be classified into two categories: specific feature based and deep learning based approaches.

The specific feature based approaches require human efforts to discover distinctive features for device identification. The methods rely on the fact that there are various manufacturing imperfectnesses in wireless devices' RF frontends. These imperfectnesses do not degrade the communication quality but can be exploited to identify each transmitter uniquely. Those features are named Physical Unclonable Features (PUF) [15], [16]). Some works assume that the statistical properties of noise or errors could uniquely profile wireless devices. In [17], the authors show that the phase error of Phase Lock Loop in transmitters can provide promising results even with low Signal-to-Noise Ratio (SNR). In [18], the authors use the error between received signals and theoretical templates and use time-frequency features to fingerprint different transmitters. In [19], the authors employ the differential constellation trace figure (DCTF) to capture the time-varying modulation error of Zigbee devices. They then develop their low-overhead classifier to identify 54 Zigbee devices.

Feature-based approaches require efforts to manually extract features or high-order statistics for different scenarios. Therefore, more effortless and versatile methods are required. Deep Neural Networks (DNNs) are frequently used as a general-purpose blackbox for pattern recognition. and can significantly reduce the hardship of manual feature discovery. In [20], the authors provide a novel method that performs signal denoising and emitter identification simultaneously using an autoencoder and a Convolution Neural Network (CNN). Their solution shows promising results even with low SNR. Similar work in [21] employs a stacked denoising auto-encoder and shows similar results. DNNs perform well even on raw signals. In [22], the authors provide an optimized Deep Convolutional Neural Network to classify SDR-based emitters in 802.11AC channels, they show that, even by using raw signals without feature engineering, CNN surpasses the best performance of conventional statistical learning methods. In [23], neural networks were trained on raw IQ samples using the open dataset from CorteXlab. Their works also show similar results. Compared with specific feature based approaches, deep neural networks dramatically reduce the requirement of domain knowledge and the quality of fingerprints.

B. Incremental Learning in Deep Neural Networks

In general, DNNs are effective in non-cryptographic wireless device identification. However, a DL enabled wireless device identifier has to learn new devices' characteristics during its life cycle. Such functionalities are defined as lifelong learning or Incremental Learning (IL).

Conventionally, Transfer Learning (TL) are applied, neural networks are pretrained in the lab and then fine-tuned for deployment using specific data [24], [25]. In TL, the learning components can forget a large proportion of the knowledge they learn in the lab and adapt to new scenarios. In Incremental Learning (IL), neural networks are trained incrementally as new data come in progressively [26]. CL does not allow neural networks to forget what they have learned in the early stages compared with TL. Therefore, TL is useful when deploying new systems, and CL is useful in regular software updates and maintenance. The strategies to implement CL for DNN are in three folds:

Knowledge replay: An intuitive solution for CL is to replay data from old tasks while training neural networks for new tasks. However, such a solution requires long training time and large memory consumption. Besides, one can hardly judge how many old samples are enough to catch sufficient variations. Therefore, some studies employ generative networks or exemplars to replay data from old tasks [27]. In

[27], Generative Adversarial Network (GAN) based scholar networks are proposed to generate old samples and mixed with the current task. In this way, the deep neural network could be trained on various data without using huge memories to retain old training data [28]. However, data generators are not easy to train and retaining old data will gradually consume a lot of memory and thus not yet a good choice for wireless device identification system in IoT.

Regularization: Initially, regularization is employed to prevent models from overfitting by penalizing the magnitude of parameters [29]. In CL, regularization is employed to prevent models from changing dramatically. In this way, the knowledge (represented by weights) learned from the old tasks will be less likely to vanish when trained on new tasks. In Elastic Weight Consolidation (EWC) [30], the algorithms identify important connections and protect them from changing dramatically, in which noncritical connections are used to learn new tasks. Regularization does not require storing old samples or data generators but may not have a high accuracy as knowledge replay.

Dynamic network expansion: Network expansion strategies lock the weights of existing connections and supplement additional structures for new tasks. For instance, the Dynamic Expanding Network (DEN) [31] algorithm first trains an existing network on a new dataset with regularization. The algorithm compares the weights of each neuron to identify task-relevant units. Finally, critical neurons are duplicated and to allow network capacity expansion progressive. The problem for the method is the need to know the task information to select appropriate data flow paths.

Incremental learning, especially memoryless class incremental, is rarely covered in signal intelligence systems, such as wireless device identification, thus motivating our research.

III. Methodology

A. Zero-bias Deep Neural Network for Wireless Device Identification

We focus on deriving a protocol-agnostic wireless device identification system with incremental learning capability. Suppose that the radio signal from a specific device i, is denoted as $\hat{m}_i(t)$:

$$\hat{m}_i(t) = m_i(t) + \delta_i(t) = I(t) + \boldsymbol{i} \cdot Q(t) \tag{1}$$

where $m_i(t)$ is the message while the residual, $\delta_i(t)$, is exploited to recognize a wireless device. $\delta_i(t)$ is also defined as the pseudo noise signal. If $\delta_i(t)$ is uncorrelated with messages $m_i(t)$, the recognition algorithm can be protocol-agnostic. In this work, we use a Software-Defined Radio (SDR) receiver (USRP B210) for signal reception, therefore, $I(t)$ and $Q(t)$ are the in-phase and quadrature components respectively.

Suppose $m_i(t)$ is successfully extracted from $\hat{m}_i(t)$ and we also extract the frequency domain features from the pseudo noise as:

$$\delta_i(\omega) = FFT[\hat{m}_i(t)] - FFT[m_i(t)] \tag{2}$$

where $r_i(t)$ is the reconstructed rational baseband signal. Please be noted that $\hat{m}_i(t)$ is complex-valued (QPSK) while $r_i(t)$ can be real-valued (2FSK, 2PSK, and etc.).

We convert $\delta_i(\omega)$ into a magnitude sequence ($||\delta_j(\omega)||$), namely, Mag.-Freq. residuals, and a phase sequence ($\angle\delta_i(\omega)$), namely Phase-Freq. residuals, respectively. These three types of signal features are passed through a Deep Neural Network based wireless device identification model, depicted in Figure 1.

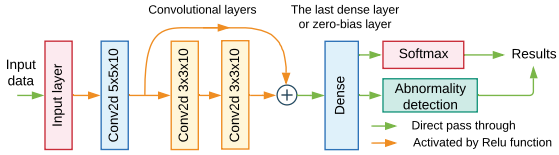


Fig. 1. Deep neural network for wireless device identification.

We have discovered that the last dense layer of a DNN classifier performs the nearest neighbor matching with biases and preferabilities using cosine similarity, We also show that a DNN classifier's accuracy will not be impaired if we replace its last dense layer with a zero-bias dense layer [32], in which the decision biases and preferabilities are eliminated. We can denote its mechanism as (also in Figure 2):

$$\begin{aligned} \boldsymbol{Y_1}(\boldsymbol{X}) &= \boldsymbol{W_0}\boldsymbol{X} + \boldsymbol{b} \\ \boldsymbol{Y_2}(\boldsymbol{X}) &= cosDistance(\boldsymbol{Y_1}, \boldsymbol{W_1}) \end{aligned} \tag{3}$$

Where $\boldsymbol{X}$ is the output of the prior convolution layers, a.k.a., feature vectors. $\boldsymbol{X}$ is an N_0 by q matrix, where N_0 denotes the number of features while q denotes the batch size. $\boldsymbol{W}_0$ is an N_1 by N_0 matrix where N_1 denotes the number of embedded features. $\boldsymbol{W_1}$ is a matrix to store fingerprints of different classes, namely the similarity matching layer and it is a C by N_1 matrix in which C denotes the number of classes, we set $N_1 = 2C$ in this paper. Please be noted that in $\boldsymbol{W_1}$, each row represents a fingerprint of corresponding class whilst in $\boldsymbol{Y}_1$ each column represents a feature vector in the latent space. Intuitively, the last dense layer is spitted into two layers, L_1 for feature embedding and L_2 for similarity matching. The cosine similarity matching is denoted as:

$$cosDistance(\boldsymbol{Y_1}, \boldsymbol{W_1}) = \boldsymbol{RU}(\boldsymbol{W_1}) \times \boldsymbol{CU}(\boldsymbol{Y_1}) \tag{4}$$

Where $\boldsymbol{RU}(\cdot)$ and $\boldsymbol{CU}(\cdot)$ denote deriving column-wise and row-wise direction vectors (vectors' magnitudes are normalized to one) of their inputs. Our prior results [32], [33] prove that the zero-bias dense layer can work seamlessly with backpropagation mechanisms and trained using regular loss functions (e.g., binary crossentropy, etc.). Please be noted that

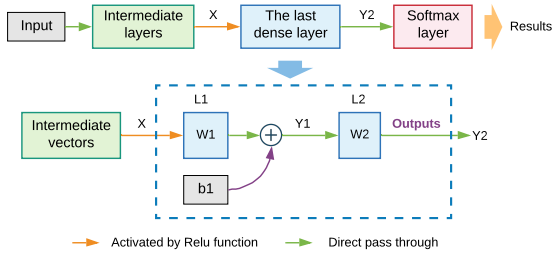


Fig. 2. Data flow of zero-bias deep neural networks.

even if L_2 can be replaced by a regular dense layer, it can also be viewed as a similarity matching layer, but the matching results are weighted and biased [32].

B. Optimal separation of fingerprints

Intuitively, if the devices' fingerprints are distantly separated in the latent space, we will have less chance to confuse them. To quantify the separation, the sum of the mutual cosine distances of all devices' fingerprints in a classification model can be defined as:

$$TD(\boldsymbol{f_1}, \cdots, \boldsymbol{f_C}) = \sum_{i=1, j<i}^{C} CosineDistance(\boldsymbol{f_i}, \boldsymbol{f_j}) = \sum_{i=1, j<i}^{C} x_i^{(1)} x_j^{(1)} + x_i^{(2)} x_j^{(2)} + \cdots + x_i^{(N1)} x_j^{(N1)} \quad (5)$$

where $\boldsymbol{f_i} = (x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(N1)})$ and $\boldsymbol{f_j} = (x_j^{(1)}, x_j^{(2)}, \cdots, x_j^{(N1)})$ are devices' fingerprint vectors. Suppose we have C devices with N_1-D fingerprint vectors. Noted that the fingerprints have been normalized into unit vectors. Therefore, if we need to find the optimal value of $TD(\cdot)$, we need to incorporate the constraints:

$$\forall i,\ g(\boldsymbol{f_i}) = \textstyle\sum_{d=1}^{N1} (x_i^{(d)})^2 - 1 = 0 \quad (6)$$

Equation 5 has now become a constrained optimization problem. We solve this constrained optimization problem with the Lagrange Multiplier as:

$$L(\boldsymbol{f_1}, \cdots, \boldsymbol{f_C}, \lambda_1, \cdots, \lambda_C) = TD(\boldsymbol{f_1}, \cdots, \boldsymbol{f_C}) - \textstyle\sum_{i=1}^{C} \lambda_i g(\boldsymbol{f}_i) \quad (7)$$

And we need to solve:

$$\nabla_{x_1^{(1)} \cdots x_1^{(N1)}, \cdots, x_C^1 \cdots x_C^{N1}, \lambda_1 \cdots \lambda_i} L(\boldsymbol{f_1} \cdots \boldsymbol{f_C}, \lambda_1 \cdots \lambda_C) = 0 \quad (8)$$

Which results in a linear system of equations. For each kth ($k = 1 \cdots N_1$) dimension of fingerprint vectors $x_1^{(k)}, \cdots, x_C^{(k)}$, we have:

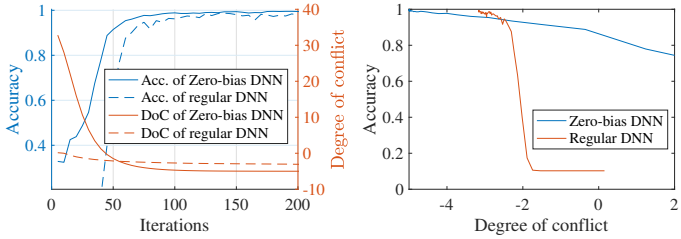
$$\begin{aligned} \tfrac{\partial L}{x_1^{(k)}} &= -2\lambda_1 x_1^{(k)} + \textstyle\sum_{i=1, i\neq 1}^{C} x_1^{(i)} = 0 \\ &\vdots \qquad\qquad \cdots \qquad\qquad \vdots \\ \tfrac{\partial L}{x_C^{(k)}} &= -2\lambda_C x_C^{(k)} + \textstyle\sum_{i=1, i\neq C}^{C} x_C^{(i)} = 0 \end{aligned} \quad (9)$$

This is a homogeneous system of equations, and it is unlikely that it only has a trivial solution (zeros). Hence, $\lambda_1 = \lambda_2 = \cdots = \lambda_C = -0.5$ and Equation 9 can be converted into one equation:

$$\textstyle\sum_{i=1}^{C} x_i^{(k)} = 0 \quad (10)$$

We square Equation 10 and expand it. According to Multinomial Theorem [34] we have:

$$\textstyle\sum_{i=1}^{C} (x_i^{(k)})^2 + 2 \sum_{n=1, m<n}^{C} x_n^{(k)} x_m^{(k)} = 0 \quad (11)$$



(a) (b)

Fig. 3. A comparison of regular and zero-bias DNN considering degree of conflicts and training accuracy.

Given that $k = 1 \cdots N_1$, we have N_1 Equations with an identical form of Equation 11. By summing them up, we have:

$$\sum_{k=1}^{N1} \sum_{i=1}^{C} (x_i^{(k)})^2 + 2 \sum_{k=1}^{N1} \sum_{n=1, m<n}^{C} x_n^{(k)} x_m^{(k)} = 0 \quad (12)$$

On the left of Equation 12, the first part is the sum of the magnitude of fingerprint vectors. And its value is C. The second part is exactly two times $TD(\boldsymbol{f_1}, \cdots, \boldsymbol{f_C})$ in Equation 5. Therefore, we have:

Remark 1. *The sum of the mutual cosine distances of classes' fingerprints of the zero-bias DNN at a converging point is a predictable constant:*

$$TD(\boldsymbol{f_1}, \cdots, \boldsymbol{f_C}) = -\frac{C}{2} \quad (13)$$

When such a value is reached, the separation of fingerprints are maximized in the latent space, indicating the lowest degree of conflict. We will use the term *Degree of Conflict (DoC)* to describe the characteristic of the zero-bias DNN. Noted that the range of DoC is from $-\frac{C}{2}$ to $\frac{C(C-1)}{2}$. The maximum value is reached when all fingerprints collide into one single vector.

To demonstrate the Remark 1, we use a simple DNN [35] with two configurations. In the first configuration, a regular dense layer is applied for the final classification. And in the second configuration, the last dense layer is modified to perform the cosine similarity matching as in Equation 3. The two models are trained on the hand-written digit dataset (MNIST). And the change of DoC and accuracy during training are depicted in Figure 3. In Figure 3a, the degree of conflict of zero-bias DNN model converges to the predicted optimal constant $-\frac{10}{2} = -5$. However, in the regular DNN model, the metric stops at a nonoptimal point, -3. Notably, higher accuracy could sometimes reflect a lower DoC between fingerprints. Figure 3b also reveals that the zero-bias DNN model is less sensitive to the variation of DoC.

C. Analyzing the catastrophic forgetting from the conflict of fingerprints

With the cosine similarity matching mechanism. One may assume that incremental learning can be performed by simply inserting new fingerprints. However, we discover that such an intuitive method could cause significant performance degradation. An important factor to cause the performance degradation is the conflict of fingerprints.

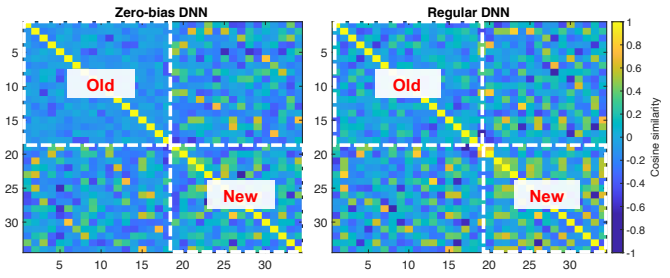


Fig. 4. Distance matrix of fingerprints after inserting new fingerprints and finetuning.

To exemplify this phenomenon, we use two DNN models with an architecture specified in Figure 1, we modify their last dense layers as in Figure 2, we use cosine similarity matching in L_2 for the first DNN model and use regular dense layer for L_2 for the second one, and therefore, the second DNN is a regular DNN. The two models are tested using a two-stage incremental learning scheme: a) in the first learning stage, the two models are first trained on a wireless signal identification dataset [36] to classify 18 most frequently seen wireless devices. b) Before the second learning stage, we insert the hypothetic fingerprints (generated by averaging feature vectors) of the remaining 16 new devices into their similarity matching layers and freeze all prior layers and fingerprints of learned devices. c) In the second stage, the IL stage, we finetune the newly inserted fingerprints. After the two-stage learning, the cosine similarity matrix of fingerprints in the two models before and after incremental learning is compared in Figure 4.

The results in Figure 4 indicate a typical conflict scheme. On the one hand, some fingerprints of the newly learned classes (devices) are less distantly separated as they have higher cosine similarities. On the other hand, some new devices' fingerprints have high cosine similarities with old devices' fingerprints. These two factors jointly cause conflict and confusion. A more detail comparison is provided in Table I. The two models degrees of maturity after IL are far from the expected optimal value. And DoC of the new fingerprints are also far from optimal.

TABLE I
A COMPARISON OF THE DEGREE OF MATURITY OF DNN MODELS BEFORE AND AFTER INCREMENTAL LEARNING.

DNN models	DoC (Acc.) initial training	DoC (Acc.) a.f.t. finetuning	DoC of new fingerprints	Acc. on new / old task
Regular	-8.083 (90.54)	-1.16 (65.2)	9.05	75.5 / 54.2
Zero-bias	-8.96 (92.85)	-4.3 (84.2)	4.03	76.2 / 91.3
Optimal value	-9	-18 (92.2)	-8	92.2 / 93.1

Interestingly, the zero-bias DNN outperforms the regular DNN considering less catastrophic forgetting, herein, we will implement our incremental learning algorithm based on zero-bias DNN. Even though zero-bias DNN has an advantage in IL, we claim that:

Remark 2. *Without readjustment of old fingerprints or proper separation between old and new fingerprints, the conflict between fingerprints can not be resolved. Under such criteria, the resulting DNN's performance will not be comparable to training with all data from scratch.*

Proof. Suppose that we have N_1 classes at the initial stage and m new classes to learn afterwards. We define that the averaged cosine distance between N_1 fingerprints is $\bar{D}_0$, according to Remark 1, after initial training we have:

$$\frac{N_1(N_1-1)}{2}\overline{D_0} = -\frac{N_1}{2} \text{ and } \overline{D_0} = -\frac{1}{N_1-1} \tag{14}$$

When we have N_1+m classes, $\bar{D}_0$ has to become:

$$\overline{D_1} = -\frac{1}{N_1+m-1} \tag{15}$$

It means that if the classes' fingerprints are to be distantly and uniformly separated, the averaged angles of all old fingerprints need to be reduced while learning new classes. This requirement can not be satisfied if the old fingerprints are locked or prevented from changing. When the prior layers are locked, the distribution of feature vectors in the latent space is fixed, simply reducing the separation of fingerprints in old classes will increase the degree of conflict and cause performance degradation, as depicted in Figure 3b. □

And if N_1+m gets larger, $\overline{D_1}$ will approximate zero, thus the averaged separation angles between fingerprints should approximate 90 degrees, that is, orthogonal. Therefore, we believe that there will potentially be some improvement if we can properly separate the fingerprints of old and new classes into different topological spaces to avoid conflict.

D. Channel separation enabled incremental learning

To resolve the conflict of fingerprints, we proposed the Channel Separation Enabled Incremental Learning (CSIL), an integral approach incorporating dimension expansion and channel separation as depicted in Figure 5. Intuitively, the merits of this approach are: a) we let the fingerprints learned at different stages to automatically use their task-specific proportions (channels) of parameters in the feature embedding layer. b) We control the directions of fingerprints and force the fingerprints learned from different stages to be orthogonally separated.

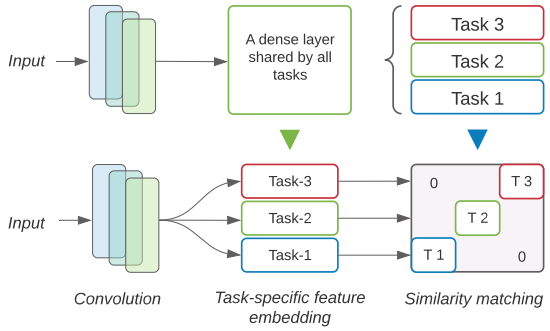


Fig. 5. Channel separation for incremental learning

At the initial stage, namely *stage-0*, we train a zero-bias DNN as normal. When at the kth learning stage, *stage-k*. We first expand the feature embedding layer's weight matrix as:

$$\boldsymbol{W_0^{(k)}} = \left[\begin{array}{c:c} \boldsymbol{W_0^{(k-1)}} & \boldsymbol{w_0^{(k)}} \end{array} \right]^T \tag{16}$$

Where $\boldsymbol{W}_0^{(k-1)}$ is the weight matrix of the feature embedding layer in $(k-1)$th stage and $\boldsymbol{w}_0$ is the expanded proportion for the kth task. We then expand the similarity matching layer of the network as:

$$\boldsymbol{W_1^{(k)}} = \left[\begin{array}{c:c} \boldsymbol{W_1^{(k-1)}} & \boldsymbol{0} \\ \hdashline \boldsymbol{0} & \boldsymbol{w_1^{(k)}} \end{array} \right] \tag{17}$$

Where $\boldsymbol{W_1^{(k-1)}}$ is the weight matrix of the similarity matching layer at in the $(k-1)$th stage and $\boldsymbol{w}_1$ is the fingerprints for the kth task. The manually inserted zeros on the one hand keep the fingerprints in different stages orthogonal (depicted in Figure 6), on the other hand, they enable the feature embedding layer to learn task-specific parameters in different regions (a.k.a. channels). For instance, in Equation 17, the newly inserted fingerprints in $\boldsymbol{w_1^{(k)}}$ only make use of embedded features from $\boldsymbol{w_0^{(k)}}$ in Equation 16.

We only train the network with data from the kth stage, we use Knowledge Distillation (KD [37]) and Elastic Weight Consolidation (EWC [30]) to prevent the model from forgetting. Therefore, the loss function is defined as:

$$L(\boldsymbol{\Theta_{k-1}}, \boldsymbol{\theta_k}, \boldsymbol{G_m}, X_k) = L_{CE} + L_D + L_{EWC}$$

Where $\boldsymbol{\Theta_{k-1}}$ denotes the models' weight at the $(k-1)$th stage. And $\boldsymbol{\theta_k} = \{\boldsymbol{w_0^{(k)}}, \boldsymbol{w_1^{(k)}}\}$ denotes the extended weights for the kth stage. $\boldsymbol{G_m}$ is a mask matrix, in which the value of each element can only be zero or one. These elements are one-to-one bound to the parameters of a neural network to control which parameter is locked or unlocked. X_k is the training data of the kth stage. L_{CE} is the cross entropy loss. L_D is the Knowledge Distillation loss:

$$L_D = \|\boldsymbol{R_{k-1}(X_k)} - \boldsymbol{R_k(X_k)}\| \tag{18}$$

Where $\boldsymbol{R_{k-1}(X_k)}$ is the response of $(k-1)$th model on X_k and $\boldsymbol{R_k(X_k)}$ is the response of the kth model. $\boldsymbol{F(\cdot)}$ denotes the output of the similarity matching layer (L_2). Knowledge Distillation aims to penalize DNNs' behavior from changing drastically.

L_{EWC} in Equation 18 represents the Elastic Weight Consolidation (EWC) loss. In EWC, Fisher Information Loss is used to measure the importance of existing parameters, we define EWC Loss for incremental learning as:

$$L_{EWC}(\boldsymbol{\Theta_k}) = \frac{1}{2}\sum_i [\boldsymbol{F_{k-1}} \cdot (\boldsymbol{\Theta_k} - \boldsymbol{\Theta_{k-1}})^2] \tag{19}$$

Where $\boldsymbol{F_{k-1}}$ denotes the Fisher Information (FI) matrix with respect to the $(k-1)$th task. Intuitively, this loss function penalizes the change of critical parameters. The matrix can be estimated as:

$$\begin{aligned} \boldsymbol{F_{k-1}} &= \left[\frac{\partial \log P(X_{k-1}|\boldsymbol{\Theta_{k-1}})}{\partial \boldsymbol{\Theta_{k-1}}} \right]^2 \\ P(X_{k-1}|\boldsymbol{\Theta_{k-1}}) &\approx \overline{\boldsymbol{Y}_{Softmax}}(X_{k-1}|\boldsymbol{\Theta_{k-1}}) \end{aligned} \tag{20}$$

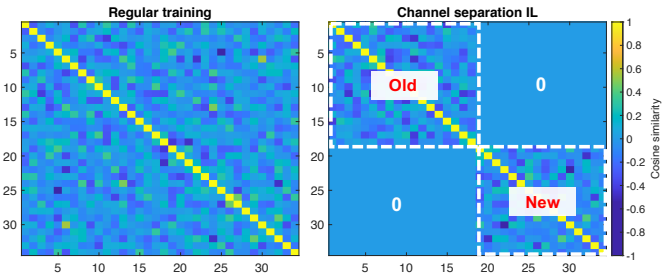


Fig. 6. Distance matrix of fingerprints after regular training and CSIL.

Where $\overline{\boldsymbol{Y}_{Softmax}}(X_{k-1}|\boldsymbol{\Theta_{k-1}})$ denotes the averaged outputs of Softmax layer on validation set X_{k-1} given parameter set $\boldsymbol{\Omega}$, it approximates the posterior probability $P(X_{k-1}|\boldsymbol{\Theta_{k-1}})$. $\boldsymbol{F_{k-1}}$ denotes the Fisher information matrix.

To exemplify the concept, in Figure 6, we compare the fingerprints' cosine similarity matrix after regular training and CSIL using the same dataset and scheme specified in Section III-C. In this experiment, the convolution layers, the channels for the old task in the feature embedding layer, and the manually supplemented zeros in the similarity matching layer are locked. As a comparison, the DoC of fingerprints is much less apparent compared to Figure 4. A more systematic comparison is provided in Section IV.

IV. Performance Evaluation

In this section, we will evaluate the performance of CSIL algorithm and compare it with the state-of-art.

A. Evaluation dataset

We use real-world ADS-B signals to verify IL methods for wireless device identification. ADS-B signals are transmitted by commercial aircraft to periodically broadcast their enroute information to Air Traffic Control (ATC) Centers in plain text. These signals are easy to receive and decode but are subject to identity spoofing attacks. We configure our SDR receiver (USRP B210) with a sample rate of 8MHz at 1090MHz, and for each piece of intercepted message, we use the first 1024 complex samples. This dataset is publicly available at [38]. We first decode the ADS-B signals using a modified version of *Gr-ADS-B* in [36] to extract the payloads, then the aircraft's identity codes are used as labels for the truncated messages' signals. We filter out the wireless transponders with less than 500 samples and use the top 100 most frequently seen transponders to construct the dataset. As in Section III-A, we extract the pseudonoise supplemented with the frequency domain information, we convert each truncated message signal record into a 32 by 32 by 3 tensor. Finally, we got 100 wireless transponders. We use 60% of the dataset for training and the remaining 40% of the dataset for validation.

B. Performance comparison

In this subsection, we compare the CSIL algorithm with other incremental learning algorithms that do not require

historical data. The configurations of the selected methods are as follows:

- **Channel Separation Enabled Incremental Learning (CSIL):** We lock the convolution layers and channels in the feature embedding layer which are used by old tasks. We train the new task-specific channels and fingerprints of devices.
- **Learning without Forgetting (LwF):** We lock the convolution layers and the feature embedding layer, we use LwF to train the similarity matching layer.
- **Elastic Weight Consolidation: (EWC)**We lock the convolution layer and feature embedding layer, we train the whole similarity matching layer. The EWC algorithm can adjust old and new fingerprints simultaneously.
- **Finetuning:** We lock the convolution layer, the feature embedding layer, and the old fingerprints, we train the similarity matching layer on new fingerprints.

In these configurations, we set the initial learning rate to be 0.01, momentum to 0.9, and $L2$ regularization factor to be 0.01. Stochastic Gradient Descent is selected. We divide the data tensors from 100 wireless devices into 5 batches. We first train the selected DNN model with 20 randomly selected devices and then incrementally train the model with other data batches. During incremental training, the batch size is set to 64 and the models are trained for 10 epoches.

We compare their resulting models' performance on old and incrementally learned new devices as in Figure 8. Since no historical data is available during incremental learning, forgetting of old tasks are unavoidable. From Figure 8a, the performances of all selected IL algorithms in recognizing new devices are close to the optimal non-IL scheme, in which the proposed CSIL yields the highest accuracy after IL while finetuning with locked old fingerprints shows the worst result. Comparably, in Figure 8b, in preventing forgetting, CSIL's performance is not as good as finetuning with locked weights after learning more than 60 wireless devices (classes). Finetunning with locked weights prevents DNN models from forgetting but with a side effect that prevents the network from learning new devices. The overall performance is given in Figure 8c, our proposed algorithm CSIL yields the best performance on both old and new tasks.

A comparison of the metric, the Degree of Conflict (DoC), of all devices' fingerprints during incremental learning, is given in Figure 7. The propose method, CSIL, yields the lowest DoC. Please be noted that the models' DoC values are still lower than the optimal values (please refer to Equation 13) after incremental learning.

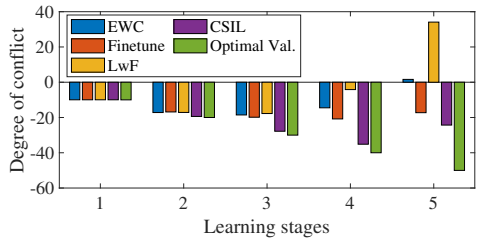


Fig. 7. Comparison of Degree of Conflict among IL algorithms

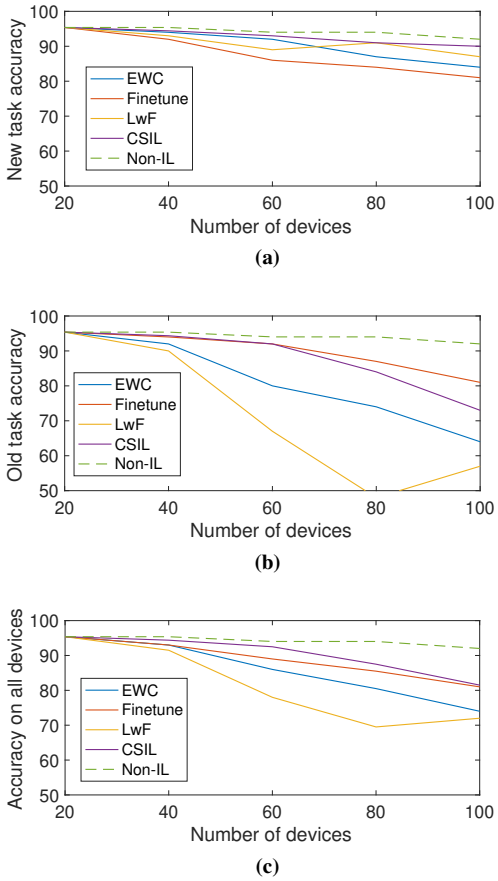


Fig. 8. Comparison of incremental learning strategies for wireless device identification

C. Ablation Analysis

We compare the averaged stage loss of the CSIL considering three factors: a) the Fisher loss. b) The Knowledge Distillation loss. c) The effect of channel separation. The results of ablation analysis are given in Table II. Apparently, the integral method combining channel separation, EWC, and Knowledge Distillation provides the best performance.

Notably, without channel separation, the combination of elastic weight consolidation and knowledge distillation can also prevent the network from forgetting. However, such a combination also prevents the network from learning new tasks. Therefore, elastic weight consolidation and knowledge distillation jointly prevent the network from forgetting old devices when training on new tasks, meanwhile, the channel

TABLE II
ABLATION ANALYSIS OF CSIL. ALL METRICS ARE IN PERCENTAGE.

Approaches	Initial Acc.[1]	Acc. with all 100 devices [2]	New acc. at the last stage	Old acc. at the last stage	Forget / stage [3]
CS + EWC + KD	95.2	**83.5**	**90**	**73**	**4.5**
~~CS~~ + EWC + KD	95.2	75.3	82.4	66.3	5.78
CS + ~~EWC~~ + KD	95.2	70.5	91	50	9
CS + EWC + ~~KD~~	95.2	70.5	91	50.2	9

[1] Identification accuracy on the first 20 devices, at this stage the network is trained from scratch.
[2] Overall accuracy (100 devices) after the last stage of incremental learning.
[3] Averaged decrease of accuracy on all trained devices after each incremental learning stage.

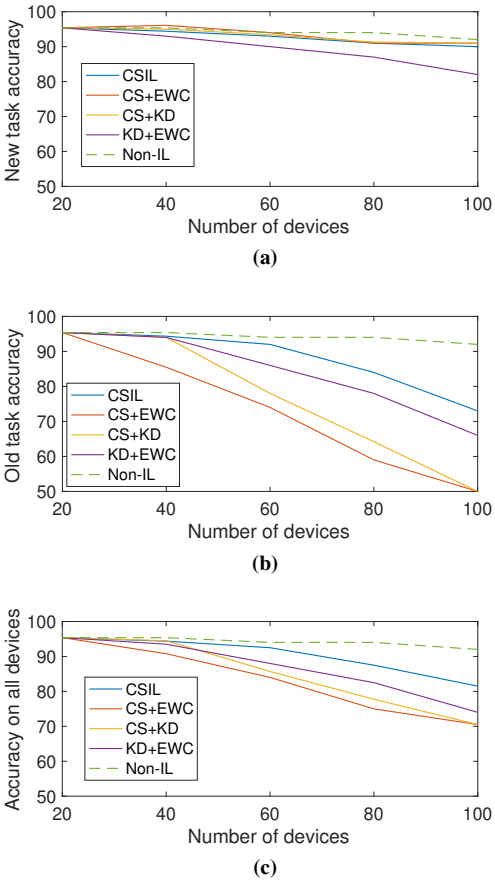


Fig. 9. Comparison influential factors in CSIL during incremental learning

separation mechanism prevents the conflict of class-specific fingerprints.

A more detailed comparison is presented in Figure 9. In Figure 9a, if the channel separation mechanism is not available, the DNN model will not perform well in learning new devices (classes), as analyzed in Remark 2, the incrementally inserted fingerprints of new devices can conflict with the existing ones, causing the performance degradation. In Figure 9b, the integral solution, CSIL, yields the highest accuracy in terms of memorizing old devices. Interestingly, the integral of knowledge distillation and elastic weight consolidation ranks the second place in memorizing old devices while showing the worst performance for learning new ones. Therefore, the CSIL provides the best balanced performance between learning and forgetting.

V. Conclusion

In this paper, we propose a novel incremental learning strategy, the Channel Separation Enabled Incremental Learning (CSIL), for wireless non-cryptographic device identification in IoT. Different from existing works, we focus on analyzing the catastrophic forgetting from a new and more thorough perspective, the conflict of device-specific fingerprints. We also propose a novel incremental learning algorithm without using historical data.

Our contributions are as follows: Firstly, we provide a new metric, Degree of Conflict (DoC), to measure the degree of topological maturity of DNN models and discover that one important cause for performance degradation in IL is the conflict of classes' representative fingerprints, in which the fingerprints of different devices (classes) are with high cosine similarity, thereby causing confusion. Second, we also show that the conventional IL schemes without using historical data, can lead to DNN models with low topological maturity and high DoC. Thirdly, based on the theoretic analysis, we propose a new IL scheme, the CSIL, based on channel separation and topological control of devices' fingerprints at different stages of learning. We evaluation our proposed solution using the raw signal records from more than 100 aircraft's wireless transponders, and the experiments demonstrate that our CSIL strategy provides the best balance between learning new devices incrementally while retaining the memory of old devices. Therefore, we believe the CSIL and the metric for quantifying the topological maturity of DNN models can be generalized to other domains, such as virus detection or medical image classification. In the future, we will focus on how to better regulate the topological space of DNN models.

Acknowledgment

This research was partially supported by the National Science Foundation under Grant No. 1956193.

References

[1] S. Jeschke, C. Brecher, H. Song, and D. Rawat, *Industrial Internet of Things: Cybermanufacturing Systems*. Cham, Switzerland: Springer, 2017.

[2] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.

[3] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli, "Fed-iiot: A robust federated malware detection architecture in industrial iot," *IEEE Transactions on Industrial Informatics*, 2020.

[4] G. Dartmann, H. Song, and A. Schmeink, *Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things*. Elsevier, 2019.

[5] H. Song, D. B. Rawat, S. Jeschke, and C. Brecher, *Cyber-physical systems: foundations, principles and applications*. Morgan Kaufmann, 2016.

[6] H. Song, R. Srinivasan, T. Sookoor, and S. Jeschke, *Smart cities: foundations, principles, and applications*. John Wiley & Sons, 2017.

[7] Z. Chu, P. Xiao, M. Shojafar, D. Mi, J. Mao, and W. Hao, "Intelligent reflecting surface assisted mobile edge computing for internet of things," *IEEE Wireless Communications Letters*, 2020.

[8] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Machine learning for the detection and identification of internet of things (iot) devices: A survey," *arXiv preprint arXiv:2101.10181*, 2021.

[9] H. Song, G. Fink, and S. Jeschke, *Security and privacy in cyber-physical systems*. Wiley Online Library, 2017.

[10] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 616–644, 2020.

[11] E. Belouadah, A. Popescu, and I. Kanellos, "Initial classifier weights replay for memoryless class incremental learning," *arXiv preprint arXiv:2008.13710*, 2020.

[12] Y. Liu, Y. Su, A.-A. Liu, B. Schiele, and Q. Sun, "Mnemonics training: Multi-class incremental learning without forgetting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 245–12 254.

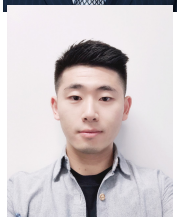
[13] X. Liu, C. Wu, M. Menta, L. Herranz, B. Raducanu, A. D. Bagdanov, S. Jui, and J. v. de Weijer, "Generative feature replay for class-incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 226–227.

[14] A. Libby and T. J. Buschman, "Rotational dynamics reduce interference between sensory and memory representations," *Nature Neuroscience*, pp. 1–12, 2021.
[15] B. Chatterjee, D. Das, S. Maity, and S. Sen, "Rf-puf: Enhancing iot security through authentication of wireless nodes using in-situ machine learning," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 388–398, 2018.
[16] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1126–1141, 2014.
[17] M. Azarmehr, A. Mehta, and R. Rashidzadeh, "Wireless device identification using oscillator control voltage as rf fingerprint," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 2017, pp. 1–4.
[18] Z. Zhuang, X. Ji, T. Zhang, J. Zhang, W. Xu, Z. Li, and Y. Liu, "Fbsleuth: Fake base station forensics via radio frequency fingerprinting," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. ACM, 2018, pp. 261–272.
[19] L. Peng, J. Zhang, M. Liu, and A. Hu, "Deep learning based rf fingerprint identification using differential constellation trace figure," *IEEE Transactions on Vehicular Technology*, 2019.
[20] J. Yu, A. Hu, F. Zhou, Y. Xing, Y. Yu, G. Li, and L. Peng, "Radio frequency fingerprint identification based on denoising autoencoders," *arXiv preprint arXiv:1907.08809*, 2019.
[21] J. Huang, Y. Lei, and X. Liao, "Communication transmitter individual feature extraction method based on stacked denoising autoencoders under small sample prerequisite," in *2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE, 2017, pp. 132–135.
[22] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep learning convolutional neural networks for radio identification," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 146–152, 2018.
[23] C. Morin, L. Cardoso, J. Hoydis, J.-M. Gorce, and T. Vial, "Transmitter classification with supervised deep learning," *arXiv preprint arXiv:1905.07923*, 2019.
[24] S. Niu, Y. Liu, J. Wang, and H. Song, "A decade survey of transfer learning (2010–2020)," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 2, pp. 151–166, 2020.
[25] S. Niu, J. Wang, Y. Liu, and H. Song, "Transfer learning based data-efficient machine learning enabled classification," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, 2020, pp. 620–626.
[26] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, 2019.
[27] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Advances in Neural Information Processing Systems*, 2017, pp. 2990–2999.
[28] G. M. Van de Ven and A. S. Tolias, "Generative replay with feedback connections as a general strategy for continual learning," *arXiv preprint arXiv:1809.10635*, 2018.
[29] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural computation*, vol. 7, no. 2, pp. 219–269, 1995.
[30] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
[31] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," *arXiv preprint arXiv:1708.01547*, 2017.
[32] Y. Liu, J. Wang, J. Li, H. Song, T. Yang, S. Niu, and Z. Ming, "Zero-bias deep learning for accurate identification of internet of things (iot) devices," *IEEE Internet of Things Journal*, 2020.
[33] Y. Liu, J. Wang, S. Niu, and H. Song, "Deep learning enabled reliable identity verification and spoofing detection," in *International Conference on Wireless Algorithms, Systems, and Applications*. Springer, 2020, pp. 333–345.
[34] "Multinomial theorem — brilliant math & science wiki," https://brilliant.org/wiki/multinomial-theorem/, (Accessed on 03/18/2021).
[35] MathWorks, "Create simple deep learning network for classification," https://www.mathworks.com/help/deeplearning/ug/create-simple-deep-learning-network-for-classification.html, May 2018.
[36] Y. Liu, J. Wang, H. Song, S. Niu, and Y. Thomas, "A 24-hour signal recording dataset with labels for cybersecurity and IoT," 2020. [Online]. Available: http://dx.doi.org/10.21227/gt9v-kz32
[37] M. Phuong and C. Lampert, "Towards understanding knowledge distillation," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5142–5151.
[38] Y. L. J. W. S. N. H. Song, "Ads-b signals records for non-cryptographic identification and incremental learning." 2021. [Online]. Available: https://dx.doi.org/10.21227/1bxc-ke87

Yongxin Liu (LIU11@my.erau.edu) received his first Ph.D. from South China University of Technology in 2018. He is a Ph.D. student in Electrical Engineering and Computer Science at Embry-Riddle Aeronautical University, Daytona Beach, Florida, and a graduate research assistant in the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). His major research interests include data mining, wireless networks, the IoT, and unmanned aerial vehicles.

Jian Wang (wangj14@my.erau.edu) is a Ph.D. student in the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, Florida, and a graduate research assistant in the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). He received his M.S. from South China Agricultural University in 2017. His research interests include wireless networks, unmanned aerial systems, and machine learning.

Jianqiang Li (lijq@szu.edu.cn) received his B.S. and Ph.D. degrees from the South China University of Technology in 2003 and 2008, respectively. He is a Professor with the College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His major research interests include Internet of Things, robotic, hybrid systems, and embedded systems.



Shuteng Niu (shutengn@my.erau.edu) is a Ph.D. student in the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University (ERAU), Daytona Beach, Florida, and a graduate research assistant in the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). He received his M.S. from ERAU in 2018. His research interests include machine learning, data mining, and signal processing.

Houbing Song (M'12–SM'14) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, in August 2012.

In August 2017, he joined the Department of Electrical Engineering and Computer Science, Embry-Riddle Aeronautical University, Daytona Beach, FL, where he is currently an Assistant Professor and the Director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us). He has served as an Associate Technical Editor for IEEE Communications Magazine (2017-present), an Associate Editor for IEEE Internet of Things Journal (2020-present), IEEE Transactions on Intelligent Transportation Systems (2021-present) and IEEE Journal on Miniaturization for Air and Space Systems (J-MASS) (2020-present). He is the editor of seven books, including Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things, Elsevier, 2019, Smart Cities: Foundations, Principles and Applications, Hoboken, NJ: Wiley, 2017, and Industrial Internet of Things, Cham, Switzerland: Springer, 2016. He is the author of more than 100 articles. His research interests include cyber-physical systems/Internet of things, cybersecurity and privacy, AI/machine learning/big data analytics, and unmanned aircraft systems. His research has been featured by popular news media outlets, including IEEE GlobalSpec's Engineering360, Association for Unmanned Vehicle Systems International (AUVSI), Fox News, USA Today, U.S. News & World Report, The Washington Times, and New Atlas.

Dr. Song is a senior member of ACM and an ACM Distinguished Speaker. Dr. Song was a recipient of 5 Best Paper Awards (CPSCom-2019, ICII 2019, ICNS 2019, CBDCom 2020, WASA 2020).