Intelligent Signal Classification in Industrial Distributed Wireless Sensor Networks-Based IIoT

Mingqian Liu, *Member, IEEE,* Ke Yang, Nan Zhao, *Senior Member, IEEE,* Yunfei Chen, *Senior Member, IEEE,* Hao Song, and Fengkui Gong, *Member, IEEE*

***Abstract*—In industrial sensor networks, complex industrial environments may be encountered leading to a mix of signals of different types. Complicated interference caused by mixed signals on industrial equipments may significantly degrade the classification rate of signals, which may result in a long training time in order to extract features. In addition, with limited channel resources, it is difficult to make the global optimal decision in industrial distributed wireless sensor networks (IDWSN). To address this problem, a signal classification method using feature fusion is proposed for industrial Internet of things (IIoT) in this paper. In the proposed method, the received signals of nodes are processed by frequency reduction and sampling pretreatment, based on which intelligent representations of signals are obtained. Using federated learning, the data samples are trained with the feature fusion network. Moreover, the trained deep learning network is used on each sensor node to classify signals, the results of which will be transmitted to aggregation center. In the aggregation center, the improved evidence theory method is used to aggregate the recognition results of each sensor node to achieve the final classification. Simulation shows that the proposed method has excellent classification performances. Notably, it is not required for the proposed method to transmit signals from nodes to the aggregation center, which could effectively protect the privacy of industrial information.**

***Index Terms*—Industrial Internet of things, industrial distributed sensor networks, signal classification, feature fusion, federated learning, alpha-stable noise**

I. Introduction

INDUSTRIAL internet of things (IIoT), integrate Internet, new-generation information technology and industrial system. In IIoT, a various of entities and front-ends, such as machines, raw materials, control systems, information systems, products and people, are interconnected. A number of advanced technologies are needed to facilitate efficient and reliable IIoT networks, including the comprehensive deep perception of industrial data, real-time transmission and exchange, rapid computing process and modeling analysis for intelligent control, operation optimization and production organization mode change [1]-[2]. The birth of industrial distributed wireless sensor networks (IDWSN) technology has further promoted the development of IIoT. IDWSN is able to overcome many problems in traditional field bus technologies and industrial Ethernet technologies. Furthermore, the significant advantages of IDWSN, such as massive nodes, the wide-range distribution, multiple data transmission paths, and simple network layout, result in a wide range of applications [3]-[5].

One key and enabling technology of IDWSN is wireless signal classification, which is used to identify the modulation information of wireless signals. Wireless signal classification is essential in many applications, such as signal demodulation, suspicious transmission monitoring, anomaly detection, and interference localization [6]-[7]. In a distributed network, propagation and radio environments vary fast with great randomness. Even for the same signals, transmissions at different times may generate different signals observed at a receiving sensor. Moreover, since all users operate in a distributed fashion without centralized control, interference in such an environment will become uncontrollable and unpredictable. The artificial noise, such as alpha-stable noise, will worsen this issue by causing considerable interference to industrial signals. To achieve the global optimal hypothesis testing, it is required that the complete observation data are transmitted to the primary node without loss. However, the limitation of channel capacity in sensor networks makes this not possible [8]. To cope with this local signal processing can be utilized in a complex industrial environment for the purpose of effective signal classification out of mixed signals with various modulation methods, frequent parameter changes, and strong interference. Artificial intelligence has been widely used in signal classification [9]-[10]. The signal classification leveraging artificial intelligence has been proven effective to address the problems of signal classification.

Deep learning, as a powerful and efficient artificial intelligence technology, has unique technical advantages and could bring in significant performance gains in many application and fields, such as accuracy, efficiency, and robustness [11]-[13]. With deep learning, complex machine learning algorithms can be efficiently realized with the preferable performance. Recently, deep learning has been utilized to develop electro-

This work was supported in part by the National Natural Science Foundation of China under Grant 61501348, Grant 61801363 and Grant 61871065, in part by the Shaanxi Provincial Key Research and Development Program under Grant 2019GY-043, in part by the China Postdoctoral Science Foundation under Grant 2017M611912, in part by the 111 Project under Grant B08038, and in part by the China Scholarship Council under Grant 201806965031. *(Corresponding author: Nan Zhao.)*

M. Liu, K. Yang and F. Gong are with the State Key Laboratory of Integrated Service Networks, Xidian University, Shaanxi, Xi'an 710071, China (e-mail: mqliu@mail.xidian.edu.cn; kyang1224@stu.xidian.edu.cn; fkgong@xidian.edu.cn).

N. Zhao is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, P. R. China. (e-mail: zhaonan@dlut.edu.cn).

Y. Chen is with the School of Engineering, University of Warwick, Coventry, West Midlands United Kingdom of Great Britain and Northern Ireland CV4 7AL (e-mail: Yunfei.Chen@warwick.ac.uk).

H. Song is with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, 24060, USA (e-mail: haosong@vt.edu).

magnetic signal recognition methods. In [14], the bispectrum estimation of the electromagnetic signal and the sparse self encoder were applied in identifying signals. However, this method cannot guarantee the global optimization in the considered model. In [15], a signal recognition method was designed based on the wavelet fuzzy neural network method to address the problem caused by various types of noise. Unfortunately, this method does not perform well in convergence speed and is easy to end in local extremum. In [16], the performance of different deep reinforcement algorithms in resource allocations is discussed. Most existing classification algorithms depend on a strong assumption of the Gaussian white noise, which is normally unrealistic in industrial applications with complicated and mixed signals. With alpha-stable noise, the modulation classification of signal is very challenging. In [17], a joint estimation algorithm based on the generalized cyclic spectrum is proposed. In [18], a new method based on explicit countless cost function and global optimization is designed. The authors in [19] put forward a modulation type classification method using sparse signal decomposition (SSD) of additive mixture Gaussian noise and impulse noise with an over complete mixture dictionary.

In an environment of industrial distributed networks, the main challenge exists in the training of learning model, where nodes have to transmit data for global optimal classification performance. To support distributed learning system, enormous resources are required for signal processing and data exchange. However, due to the scarcity of channel resources, how to use limited resources to effectively execute distributed learning is a key issue. Different from the existing methods, where data need to be delivered from nodes to a main server for training and learning, we will use distributed data that collected and stored on multiple edge nodes to train machine learning model as federated learning. With federated learning, the privacy of users' personal data can be protected, and the signal classification could be improved by deep learning technologies. In view of these tremendous benefits, federated learning has been deemed as a promising technology in IDWSN signal modulation classification, vehicular Internet of Things and other fields [20].

In this paper, a novel wireless signal classification framework leveraging federated learning is proposed. The main contributions of this paper are summarized as follows:

- We propose a feature fusion network, where the deep information of intelligent representation is fused with the shallow information at different levels. The deep feature fusion structure will also be used as the input of the shallow feature fusion structure, expanding the amount of feature information. The shallow feature can encompass part of the deep feature information for the recognition accuracy improvement.
- The neural network based on feature fusion is developed for the signal detection of distributed sensors. Compared with existing methods, this structure can automatically learn and extract features. Moreover, the classification performance can be improved without affecting the transmission performance.
- Federated learning is employed for learning and recognition in distributed networks, which solves the problem of bandwidth limitation and protects the privacy of data. D-S evidence theory is adopted to aggregate the recognition for classification performance enhancement.

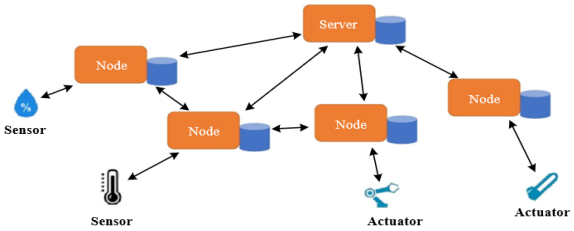

Fig. 1: System model of signal classification in IDWSN-based IIoT.

The rest of this paper is organized as follows. The system model of signal classification for IDWSN is shown in the Section II. In Section III, the intelligent representation of electromagnetic signals is presented. Federated learning on distributed sensors is developed in Section IV. In section V, specific experiments are given to verify the classification performance. Finally, Section VI show the main research findings of this paper.

II. System Model

As shown in Fig. 1, system model of signal classification in IDWSN-based IIoT is considered in this paper, which consists of one transmitter and multiple parallel sensor nodes. In an IDWSN, wireless signals are broadcast by transmitters and sent on parallel channels that experience independent channel noise. In order to reduce channel resource consumption and protect privacy, signals need to be processed and classified locally on sensor nodes, and the wireless signals $s(k)$ received on distributed sensor nodes can be expressed as

$$\mathrm{s}(k) = x(k) + e(k), k = 0, 1, ..., N-1, \tag{1}$$

where $e(k)$ represents artificial noise in IDWSN, which can be described as the alpha-stable noise. Alpha-stable noise is expressed as characteristic function:

$$\varphi(t) = \exp\{j\delta t - \gamma|t|^{\alpha}[1 + j\beta \mathrm{sgn}(t)\omega(t,\alpha)] \tag{2}$$

where

$$\mathrm{sgn}(t) = \begin{cases} 1, & t > 0, \\ 0, & t = 0, \\ -1, & t < 0, \end{cases} \tag{3}$$

$$\omega(t,\alpha) = \begin{cases} \tan(\alpha\pi/2), & \alpha \neq 1, \\ (2/\pi)\log|t|, & \alpha = 1, \end{cases} \tag{4}$$

$0 < \alpha \leq 2$ stands for the characteristic index, $\gamma \geq 0$ is the dispersion parameter and $-1 \leq \beta \leq 1$ is the index of skewness, δ represents the location parameter. $x(k)$ are wireless signals, which are comprised of amplitude modulation (AM) signal, frequency modulation (FM) signals, binary phase shift keying (BPSK) signals, quadrature phase shift keying (QPSK) signals, 8 phase shift keying (8PSK) signals, 2 amplitude shift keying (2ASK) signals, 4 amplitude shift keying (4ASK) signals, 2

frequency shift keying (2FSK) signals and 4 frequency shift keying (4FSK) signals.

In the analog modulation type, the modulation signal can be expressed as

$$s(t) = A\cos[2\pi f_c t + \phi(t) + \phi_0], \tag{5}$$

where A represents the instantaneous amplitude of the signal, f_c stands for the carrier frequency, ϕ_0 denotes the modulation phase and θ is the carrier initial phase. AM modulation signal can be expressed as

$$A = m_0 + m_t, \tag{6}$$

where m_t denotes the baseband modulation signal and m_0 represents the DC component. FM modulation signal is expressed as

$$A = 1. \tag{7}$$

For digital modulation signal, whose baseband waveform can be expressed as follows

$$s(t) = \sum_n a_n g(t - kT), \tag{8}$$

where a_n is the symbol sequence sent by the transmitter, $g(t)$ stands for the equivalent filter including shaping filter, channel filter and matching filter. Different types of modulation have the different symbol sequence renderings. MPSK signals can be written as

$$s(t) = Ae^{j(2\pi f_c t + \varphi_i)}, \tag{9}$$

where φ_i denotes the phase modulation function, which is given by

$$\varphi_i = \frac{2\pi i}{M}, i = 0, 1, ..., M-1, \tag{10}$$

MASK signals are expressed as follows

$$s(t) = Ae^{j(2\pi f_c t)}, \tag{11}$$

MFSK signals can be expressed as

$$s(t) = e^{j(w_c t + 2\pi f_i t)}, \tag{12}$$

where f_i denotes the modulation frequency.

III. Intelligent Representation of Wireless Signals

A. Generalized Envelope Square Spectrum

Since the alpha-stable noise does not have the second-order or higher-order statistics, the signal disturbed by alpha-stable noise do not have the effective envelope square spectrum. The main reason is that there is a large impulse pulse, resulting in a large amplitude in the disturbed signal, so it is necessary to preprocess the signal by

$$f(x) = \frac{(\frac{2}{1+e^{-|x(t)+jH(x(t))|}} - 1)x(t)}{|x(t) + jH(x(t))|}, \tag{13}$$

where $H(.)$ denotes the Hilbert transform. The signal of any point can be written as $x(t) = r\cos\theta$, so we can obtain

$$f(rcos\theta) = \frac{\left(\frac{2}{1+e^{-|re^{j\theta}|}} - 1\right) rcos\theta}{|re^{j\theta}|} = \left(\frac{2}{1+e^{-r}} - 1\right) cos\theta, \tag{14}$$

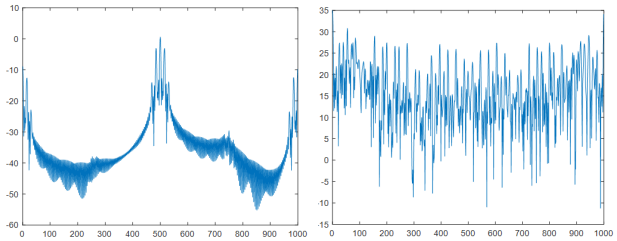

Fig. 2: Generalized envelope square spectrum of AM and FM.

where the value range of $\left(\frac{2}{1+e^{-r}} - 1\right)$ is $[-1, 1]$. Therefore, the above functions can map the amplitude of the processed signal to a range of $[-1, 1]$, and do not change the phase information of the signal.

The square of the processed signal envelope can be expressed

$$u(t) = a^2(t) = f(x)^2 + H[f(x)]^2. \tag{15}$$

Using classical spectrum estimation to estimate the power spectrum of the signal as follows

$$P(\omega) = \frac{1}{N}\left|U_N(e^{j\omega})\right|^2, \tag{16}$$

where $U_N(e^{j\omega})$ is Fourier transform of the N-point observation data of $U(t)$.

B. Fractional Lower Order Cyclic Spectrum

Fractional low-order moment is a powerful tool for analyzing and processing non-Gaussian signals. If the characteristic index of random signal is alpha, the fraction low-order moment of the signal $x(t)$ with alpha-stable noise is defined as

$$E[|X|^P] = C(p, \alpha)\gamma^{p/\alpha}, \tag{17}$$

where p represents the fractional factor, whose value range is $0 < p < \alpha \leq 2$. $C(p, \alpha)$ is a constant related to p and α. There are two methods of calculation for signal cycle spectrum based on low-order fractional moments as follows: one is based on covariance and the other is based on the low-order covariance of fractions. On the basis of fractional lower moment, the definition of p-order covariance of $x(t)$ can be expressed as

$$COV_{x,p}(t, \tau) = E[x(t + \frac{\tau}{2})x^*(t - \frac{\tau}{2})^{\langle p-1\rangle}], \tag{18}$$

where τ denotes the time delay, p stands for the order factor, and its value range is $[1, \alpha]$. α is the characteristic index of alpha-stable noise, and its value range is $[1, 2]$. If $COV_{x,p}(t, \tau)$ is a periodic function of t, it is expanded into a Fourier series, and the coefficient of the Fourier series is the fractional low-order cyclic autocorrelation function of the signal, which is expressed as

$$R^{\varepsilon}_{x,p}(\tau) = \left\langle x(t + \frac{\tau}{2})x^*(t - \frac{\tau}{2})^{\langle p-1\rangle} e^{-j2\pi\varepsilon t} \right\rangle, \tag{19}$$

where ε represents the cyclic frequency of order p. By performing the Fourier transformation of $R^{\varepsilon}_{x,p}(\tau)$, we can obtain

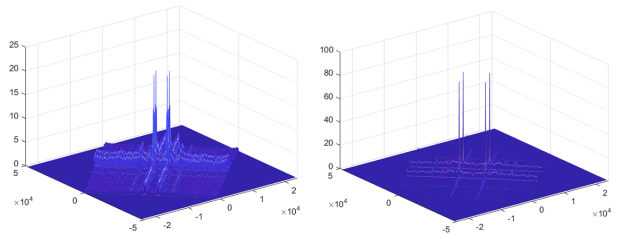

Fig. 3: Fractional lower order cyclic spectrum of 2ASK and 4ASK.

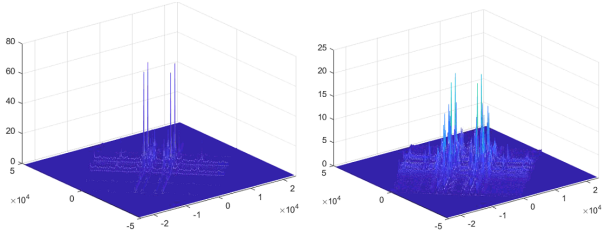

Fig. 4: Fractional lower order cyclic spectrum of 2FSK and 4FSK.

the fractional low-order cyclic spectral density function of signal $x(t)$ as

$$S_{x,p}^{\varepsilon}(f) \triangleq \int_{-\infty}^{+\infty} R_{x,p}^{\varepsilon}(\tau) e^{-j2\pi ft} d\tau, \tag{20}$$

where f stands for the normal frequency. Obviously, when the order factor $p = 2$, the fractional lower order cyclic spectral density function is the second order cyclic spectral density function.

The fractional lower order cyclic spectrum of MASK signal can be expressed as follows

$$S_x^{\varepsilon} = \begin{cases} \frac{E_b^p}{4\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2}) \\ +Q(f-f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})], & \varepsilon = m/T \\ \frac{E_b^p}{4\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2})], & \varepsilon = -2f_0 + m/T \\ \frac{E_b^p}{4\mathrm{T}}[Q(f-f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})], & \varepsilon = 2f_0 + m/T \\ 0, & other \end{cases} \tag{21}$$

and the fractional lower order cyclic spectrum of MFSK signal can be expressed as

$$S_x^{\varepsilon} = \begin{cases} \frac{E_b^p}{2\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2}) & \varepsilon = m/2T \\ +Q(f-f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})], & \\ \frac{E_b^p}{2\mathrm{T}}[-Q(f+f_0+\frac{\varepsilon}{2})\cdot & \\ Q^*(f+f_0-\frac{\varepsilon}{2})e^{-j[2\pi(\alpha+2f_0)]} & \varepsilon = \pm 2f_0 \\ -Q(f-f_0+\frac{\varepsilon}{2})\cdot & +m/2T \\ Q^*(f-f_0-\frac{\varepsilon}{2})e^{-j[2\pi(\alpha-2f_0)]}], & \\ 0, & other \end{cases} \tag{22}$$

The fractional lower order cyclic spectrum of BPSK signal is

$$S_x^{\varepsilon} = \begin{cases} \frac{E_b^p}{4\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2}) & \\ +Q(f-f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})], & \varepsilon = m/T \\ \frac{E_b^p}{4\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})e^{-2j\varphi_0} & \\ +Q(f-f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2})e^{2j\varphi_0}], & \varepsilon = \pm 2f_0 + m/T \\ 0, & other \end{cases} \tag{23}$$

and the fractional lower order cyclic spectrum of MPSK ($M \geq 4$) signals can be written as

$$S_x^{\varepsilon} = \begin{cases} \frac{E_b^p}{4\mathrm{T}}[Q(f+f_0+\frac{\varepsilon}{2})Q^*(f+f_0-\frac{\varepsilon}{2}) \\ +Q(f-f_0+\frac{\varepsilon}{2})Q^*(f-f_0-\frac{\varepsilon}{2})], & \varepsilon = m/T \\ 0, & other \end{cases} \tag{24}$$

IV. Signal Classification Based on Feature Fusion and Federated Learning in IDWSN

A. Feature Fusion Based on DenseNet

In traditional CNN layers, input x is mapped to $F(x)$, and then $F(x)$ is used to fit the target distribution. But ResNet

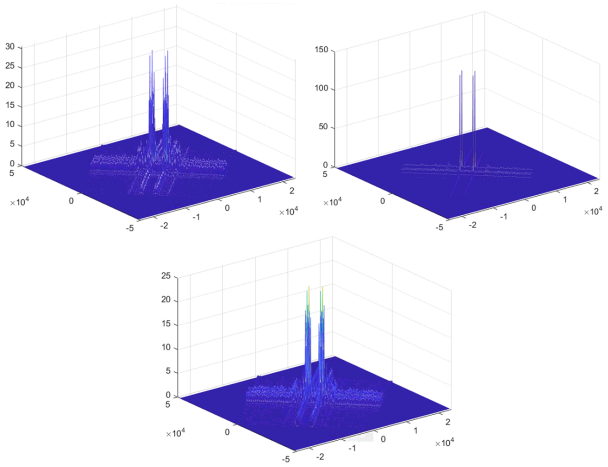

Fig. 5: Fractional lower order cyclic spectrum of BPSK, QPSK and 8PSK.

connects the input directly to the output layer through a bypass between the input and output, so that the object to be fitted by the layer changes from $F(x)$ to $G(x)$. In abstract, such a connection makes the layer only need to fit 0, which greatly reduces the problem of gradient disappearance of the deep network in the learning process. DenseNet goes further and lead into the concept of Densenlock. In a DenseBlock module, the input of each layer comes from the output of all previous layers of this layer. The basic structure of DenseNet is shown in the Fig.6.

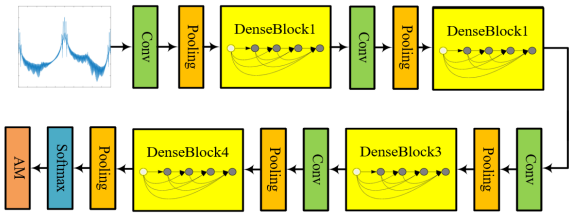

Fig. 6: Basic structure of DenseNet.

From the Fig. 6, we can see that DenseNet will first make a convolution for the input tensor with the kernel size of $[7 \times 7]$ and the step size of 2. Then make the maximum pooling with the kernel size of $[3 \times 3]$ and step size of 2. After that, there is the alternate connection between DenseBlock and Transition Layer. Finally there is a classification layer with $[7 \times 7]$ global average pooling, 1000 full connections and SoftMax. The Transition Layer is composed of BN layer, $[1 \times 1]$ convolution layer and $[2 \times 2]$ average pooling layer to reduce the number of

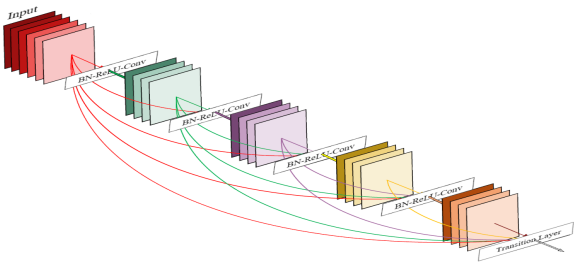

Fig. 7: Structure of DenseBlock.

feature maps. If a DenseBlock outputs m characteristic graphs and inputs the Transition Layer, then the Transition Layer will output θm characteristic graphs, where $0 < \theta \leq 1$ is the compression factor. In this paper, $\theta = 0.5$ is taken, that is, the number of channels to the next DenseBlock will be reduced by half after the transition layer compression. At the same time, in order to avoid overfitting, dropout operation is also used to reduce branches randomly. The structure of DenseBlock is shown in Fig.7, each two layers are connected by BN-ReLu-Conv, where Conv includes a $[1 \times 1]$ convolution and a $[3 \times 3]$ convolution operation.

In the traditional convolutional neural network, if the network has L layer, then there will be L connections. But in DenseBlock, the output of previous layers can be used as the input of each layer, that is, the input of lth layer comes from the output of all the previous $l-1$ layers. One of the advantages of DenseNet is that the network is narrower, the parameters are less, and a large part of the reason is due to the design of DenseBlock. In DenseBlock, the number of output feature maps of each convolution layer is exceedingly small (< 100), far less than other networks which have hundreds of feature maps. At the same time, this connection makes the transfer of features and gradients more effective, so the network is easier to train. And since each layer of DenseBlock is directly connected with input and loss, the gradient disappearance can be alleviated. In addition, this DenseBlock has the effect of regularization, which has a certain inhibitory effect on the over quasi contract samples. In this paper, four DenseBlocks are used in the networks, the first is 6 layers, the second is 12 layers, the third is 24 layers, and the fourth is 16 layers. Based on DenseNet, this paper presents a feature fusion networks, whose structure is shown in Fig. 8. The feature fusion layer is added to the original network. The feature fusion layer is mainly used to combine the deep information of input data intelligent representation with the shallow information in different degrees, so as to expand the amount of feature information, the shallow features can also contain part of the deep feature information to improve the recognition accuracy.

The structure of feature fusion layer is shown in the Fig.9. There are four feature fusion structures in the networks, which are mainly divided into three types as follows: 1) For the third feature fusion structure from left to right, input layer 1 and input layer 2 are the output of last two convolution layers. Input layer 2 is the deeper convolution layer in the whole network. Usually the feature dimension of input 2 is smaller, so we connect with the deconvolution layer to increases the

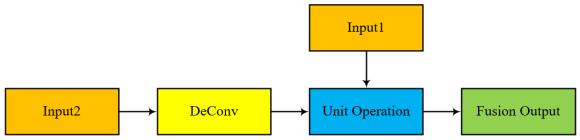

Fig. 9: Structure of feature fusion layer.

dimension size of the feature. And then the features of the two layers are processed by unit operation layer Information fusion. 2) For the first two feature fusion structures, input layer 1 is a shallow convolution layer for classification, and input layer 2 is the output layer of the previous feature fusion structure, which also passes through the deconvolution layer. Then we connect two layers through the unit operation layer, so that the shallow feature map integrates the information of multiple deep feature maps to a certain extent during prediction. 3) The last feature fusion structure is to combine the previous fusion features. In order to operate the unit, convolution or deconvolution is also needed to ensure the size of the feature. The purpose of unit operation layer is to combine feature maps which have the same size. We can point multiplied, added and subtracted, maximized or spliced the corresponding elements of the two layers. In this paper, we choose the splicing operation. The feature map is spliced according to the fourth dimension of it. The spliced feature map keeps the same size of the original map, but the number of channels is the sum of input channels. At the same time, in order to avoid too much data, $[1 \times 1]$ convolution kernel can be used to reduce the number of characteristic channels.

From the above analysis, the steps of feature fusion based on DenseNet is summarized in Algorithm 1.

Algorithm 1 The steps of feature fusion based on DenseNet.

1: The feature matrix of size $[224 \times 224 \times 2]$ is put into the feature fusion network, and set the number of convolution kernel channels K, convolution layers N in DenseBlock and compression factor θ in Transition Layer;
2: Carry on 7×7 conv with step 2 and 3×3 maxpool with step 2;
3: The output features x obtained in the second step are through the structure of DenseBlock1-Transition Layer1-DenseBlock2-Transition Layer2-DenseBl- ock3-Transition Layer3-DenseBlock4. The output of the ith Transition Layer is recorded as x_i;
4: Fuse x and x_3. Deconvolution the fusion results to the same size with x_2 and fuse them. Then deconvolution the fusion results with x_1 and fuse them to get the final fusion features;
5: $[7 \times 7]$ global average pool and $1000D$ full-connected;
6: Use SoftMax classifier to get finally result.

B. Signal Classification Based on Federated Learning

In a distributed sensor network, the unknown transmission data sequence is broadcast by the transmitter and transmitted on a parallel channel experiencing independent channel noise.

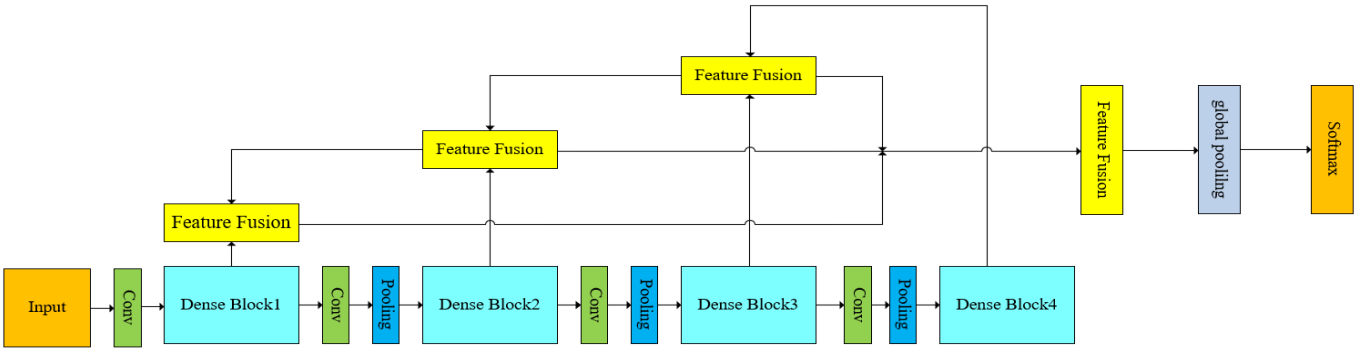

Fig. 8: Structure of feature fusion networks.

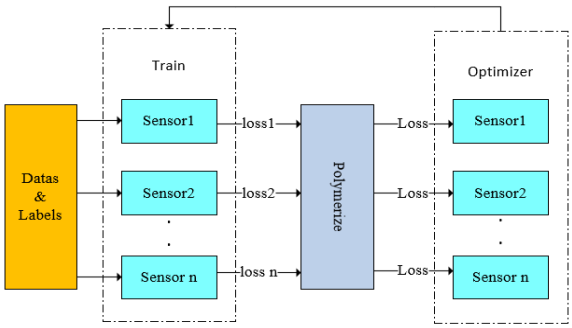

Fig. 10: Structure of federated learning in IDWSN.

It is assumed that each sensor has the same number of observations and all the sensors in the network collect and process the noisy data series at the same time. Due to the variations of propagation and transmission environments, even if the transmitter sends the same signal, different signals may be observed on the receiving sensor. To reach the global optimization of hypothesis testing, the complete sensor observation data are required to be collected by the master node.

This is a strong and unrealistic assumption, as limitation of channel capacity in sensor networks makes the master node impossible to obtain a complete set of original observations. In order to reduce the requirement of channel bandwidth, local sensors are deployed to have the relevant chip, so that the signal processing can be completed locally. The following details the federated learning on the distributed sensor network, which is divided into training and testing. The structure of federated learning process in IDWSN is shown in Fig.10.

The data is sent out by the transmitter and reaches the sensor in the network through the parallel channel. Each sensor node has set up the neural networks but has not been trained. After each sensor receives the signal, its intelligent representation is calculated and put into network training. Different from the traditional training, after getting the loss function of this training, we do not rush to the next step of gradient optimization but send the loss function of each node to the aggregator. Then aggregate the results according to some rules and return them to each node, and then each node uses the global loss function for gradient optimization to achieve the goal of global aggregation. Then it repeats until the global loss function reaches an ideal state. This can not only improve the quality of the model, reduce the channel resource occupation, but also effectively protect the privacy of the data.

The loss function and the batch size of all sensors are sent to the aggregator for aggregation and the aggregator receives the data of N sensors. According to the weighted average, the loss function after aggregation can be expressed as follows:

$$L_{loss} = \frac{\sum_{i=1}^{N} L_{loss_i} \times b_i}{\sum_{i=1}^{N} b_i}, \tag{25}$$

where L_{loss} is the loss function after aggregation, L_{loss_i} is the loss function of the ith sensor and b_i is the batch size of the ith sensor. These results are sent back to each sensor. After receiving the aggregator's return, each sensor uses the global loss function to continue to train the local network, and repeats until the loss function tends to be stable or the accuracy reaches the ideal value.

Each sensor uses gradient descent to update the network locally, then the local update in the ith sensor is carried out as follows:

$$w_i(t) = w_i(t-1) - \eta \nabla F_i(w_i(t-1)), \tag{26}$$

where $\eta > 0$ is the learning step, and the model parameter W is updated along the negative gradient direction following F_i. In [22], this method has been proved to have global convergence and good convergence performance for convex optimization problems.

From the above analysis, the training process of feature learning are given by algorithm 2.

Algorithm 2 Training process of feature learning.

1: The received signal is intelligently represented and obtain the two channel characteristic matrix;
2: The feature matrix is used to train the neural network locally. After get the cross entropy loss function of each batch, send the loss function to the master node;
3: The main node merges the loss function from each node and returns the result to each node;
4: The node uses the received loss function for gradient descent training to complete the network training.

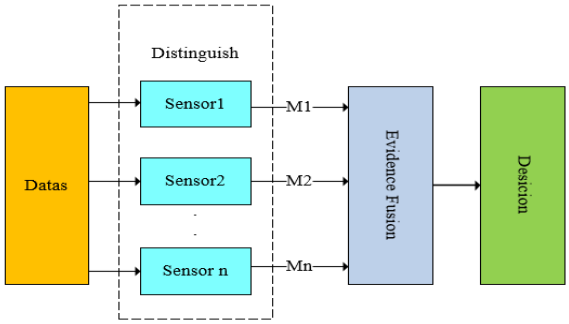

Fig. 11: Structure of Classification Fusion in IDWSN.

C. Multi-source Classification Fusion in IDWSN

The structure of the classification fusion is shown in the Fig.11. After each node receives signals, signals are classified locally by using the trained networks. We use SoftMax as the classifier, so the probability of each node's decision-making results can be obtained. And Then it is transmitted to the aggregator for aggregation to get the final classification results. This paper uses D-S evidence theory for aggregation, and we will introduce the specific steps as follows.

In this paper, we classify 9 different types of signals, so $\Theta = \{\theta_1, \theta_2, ..., \theta_9\}$ is taken as the classification framework, where $\theta_1, \theta_2, ..., \theta_9$ represent 9 signal categories, respectively. We use $m_i(\theta_j)$ to denote the probability that the signal is θ_j in the classification results of the ith sensor. Then the set vector of the independent basic probability distribution of evidence generated by the ith sensor classification result is $M_i = \{m_i(\theta_1), m_i(\theta_2), ...m_i(\theta_{12})\}$ and M_j is in the same way. Evidence theory uses evidence conflict degree to fuse evidence. The common expression methods of conflict degree include conflict coefficient, evidence distance, cosine similarity, etc. The evidence distance $dis_J(M_i, M_j)$ is usually used to describe the degree of difference between two evidences measured by the sensor as a whole, which can express as

$$dis_J(M_i, M_j) = \sqrt{0.5(M_i - M_j)^T D(M_i - M_j)}, \quad (27)$$

where $i, j = 1, 2, ..., N$, M_i, M_j are the set vectors of evidence basic probability assignment of sensor node i and j, respectively. N represents the number of the sensors. D stands for $n \times n$ matrix, which is called Jaccard coefficient and can be expressed as

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1N} \\ d_{21} & d_{22} & \cdots & d_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & \cdots & d_{NN} \end{bmatrix}, \quad (28)$$

where $d_{ij} = |A \cap B|/|A \cup B|$, $A, B \in \Theta$ and $|.|$ is used to calculate the cardinality of a set. The evidence distance $dis_J(M_i, M_j)$ also meets the following conditions

- Non-negativeness:$dis_J(M_i, M_j) > 0$;
- Non-degeneracy:$dis_J(M_i, M_j) = 0 \Leftrightarrow M_i = M_j$;
- Symmetry:$dis_J(M_i, M_j) = dis_J(M_j, M_i)$.

The trust degree between evidence can be calculated as

$$\chi_{ij} = 1 - dis_J(M_i, M_j), \quad (29)$$

so their reliability matrix is

$$\chi = \begin{bmatrix} 1 & \chi_{12} & \cdots & \chi_{1N} \\ \chi_{21} & 1 & \cdots & \chi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \chi_{N1} & \chi_{N2} & \cdots & 1 \end{bmatrix}. \quad (30)$$

From the above matrix, we can calculate that the credibility of the evidence of sensor node i to other nodes is

$$Sup(M_i) = \sum_{j=1, j \neq i}^{n} \chi_{ij}. \quad (31)$$

Normalize the credibility one by one, then we can get their respective weights as follows

$$\omega_i = \frac{Sup(M_i)}{\sum_{j=1}^{N} Sup(M_j)}. \quad (32)$$

Finally, using the above weight to fuse the evidence, the probability $m(\theta_i)$ of each classification is

$$m(\theta_i) = \sum_{j=1}^{N} \omega_j \times m_j(\theta_i) \quad (33)$$

where $M = \{m(\theta_1), m(\theta_2), ...m(\theta_{12})\}$ denotes the final classification result.

The signal classification algorithm based on federated learning in IDWSN-based IIoT is summarized in Algorithm 3.

Algorithm 3 Signal classification based on federated learning in IDWSN.

1: The received signal is intelligently represented, and obtain the two channel characteristic matrix;
2: The feature matrix is input into the trained feature fusion network for local recognition. Use SoftMax classifier to obtain the set vector of basic probability $M_i = \{m_i(\theta_1), m_i(\theta_2), ...m_i(\theta_9)\}$, and send it to the main node;
3: After the master node receives the set vector of basic probability distribution from each node, use D-S evidence theory to get the weight of each node vector;
4: Fuse set vector of basic probability distribution by weight and obtain the classification results.

The details of Algorithm 3 can be discussed as follows: main node is to receive the set vector $M_i = \{m_i(\theta_1), m_i(\theta_2), ...m_i(\theta_{12})\}$ of independent basic probability distribution of evidence generated from the classification results of each sensor, and then use (28) to obtain the trust degree between each node and form the reliability matrix shown in (29), and combine (30) and (31) to obtain the weight of each result to get the final classification results.

V. SIMULATION RESULTS AND ANALYSIS

To demonstrate the effectiveness and superiority of the proposed method, simulation experiments are conducted in this section. Nine types modulation signals are considered, including AM, FM, BPSK, QPSK, 8PSK, 2ASK, 4ASK,

TABLE I: Classification performance with the different number of DenseBlocks and revolution layers.

DenseBlocks \ Revolution layers	3	4	5	6	7	8
3	82.87%	84.91%	85.45%	87.70%	87.75%	87.84%
4	83.5%	85.4%	86.29%	88.9%	88.99%	88.47%
5	83.56%	85.57%	86.46%	88.21%	88.24%	88.77%

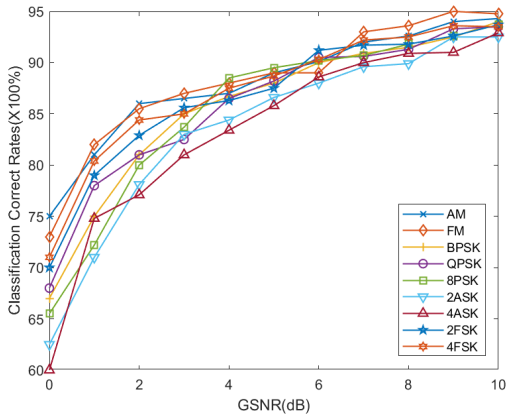

Fig. 12: Classification performance of different signals versus different GSNRs.

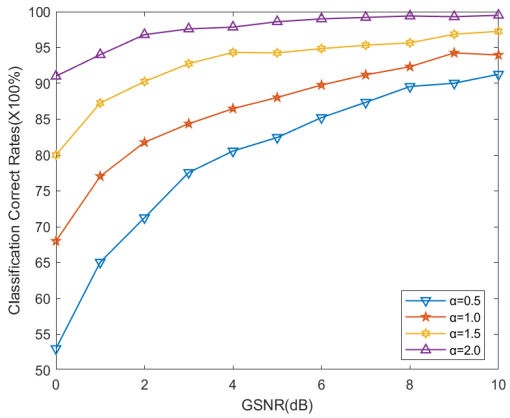

Fig. 13: Classification performance of different α versus different GSNRs.

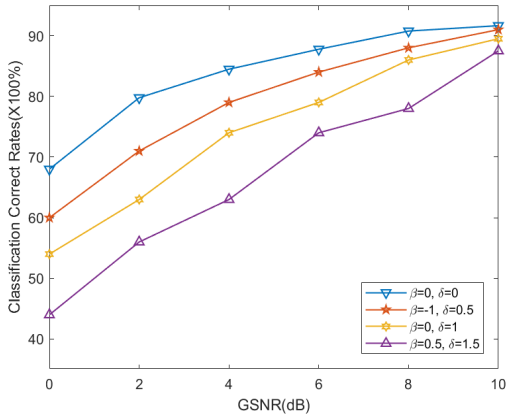

Fig. 14: Classification performance of different β and δ versus different GSNRs.

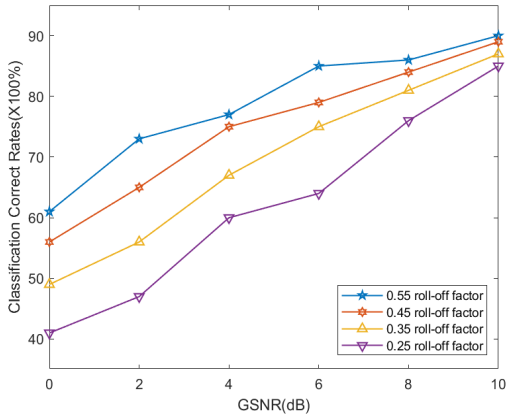

Fig. 15: Classification performance of different roll-off factors versus different GSNRs.

2FSK and 4FSK. The simulation parameters of nine types of wireless signals used in this simulation are: frequencies of 2FSK signals are 20MHz and 40MHz, frequencies of 4FSK signal are 15MHz, 25MHz, 35MHz and 45MHz, respectively. The carrier frequency of other signals is 30MHz. The pulse width is set to 10 and the sampling frequency is 120MHz. The number of samples used in training and testing of each type of signals are 10000 and 1000, respectively.

Tab. 1 shows the classification performance of signals under different DenseBlocks and revolution layers. It should be noted that the table shows the number of layers in the first DenseBlock, and the number of layers in the second, third, fourth and fifth DenseBlocks is 2, 6, 9 and 8 times of the first. From Tab. 1, when the number of DenseBlock is 4 and there are 6 layers in the first one, the classification performance tends to be stable. Increasing the number of neurons will not significantly increase the performance, or even slightly decrease.

Using the network structure with the best performance obtained above, this neural network has four DenseBlocks and the number of convolution layers of each DenseBlock is 6, 12, 36 and 54. Fig. 12 shows the classification performance of different types wireless signals versus different GSNRs. According to the Fig. 12, the average classification rates are greater than 85% when the GSNR is greater than 5 dB. When the GSNR is more than 10 dB, the classification rates can reach more than 90%, so the proposed signal classification method is effective and feasible.

Fig. 13 shows the classification performance with different alpha-stable noise characteristic parameter α values. From the Fig. 13, the classification performance will gradually improve with the increase of α. However, the classification performance improvement is exceedingly small when α is greater than 1.5. The proposed method has better classification performance for

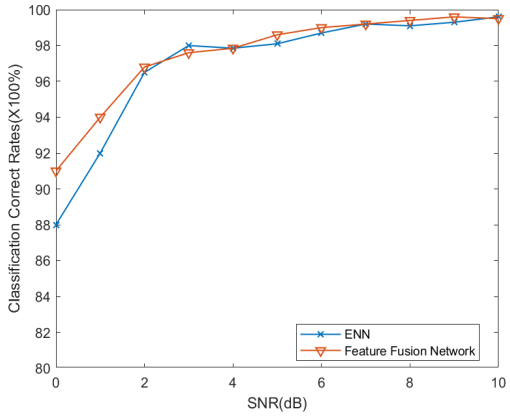

Fig. 16: Classification performance compare with different deep networks in Gaussian noise environment.

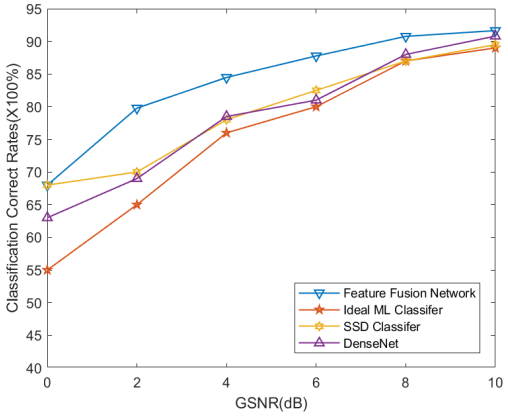

Fig. 17: Classification performance compare with different methods.

non-Gaussian noise and is also robust to the characteristic parameter α.

Fig. 14 shows the classification performance with different β and δ. From Fig.14, δ has a great impact on classification performance. With the increasing of δ, the accuracy of classification is decreasing, and the decreasing speed is faster. We also carried out the classification performance with different roll-off factors, and the results are shown in Fig. 15. From Fig. 15, we can see that the roll-off factor has a great impact on the classification performance under low GSNR, but the impact is smaller with the increasing of GSNR. When GSNR is greater than 10dB, the effect of roll off coefficient has been reduced to less than 5%.

Under the same simulation experiment environment and parameter settings, we compare the classification performance of the proposed method in Gaussian noise environment with that of the method based on ENN in [21] and the comparison results are shown in Fig. 16. From Fig. 16, we can be seen that the proposed method has better performance than that in [21] when the SNR is less than 5dB. In addition, the proposed method is suitable for signal classification in Gaussian noise environment.

In the same simulation environment and the same signal parameter settings, feature fusion network based on DenseBlocks is compared with the method in [19] and DenseNet without feature fusion with α = 1.2, and the comparison results are shown in Fig. 17. The average classification rate of the proposed method is significantly higher than other methods when the GSNR is less than 10 dB. The computational complexity of the proposed method as follows: the complexity of intelligent representation is $O(KN\log N)$, where K is the number of data segmentation; the complexity of federated learning is $O(KM)$, where K denotes the total number of global aggregation executions and M represents the number of nodes, respectively; the complexity of the feature fusion network is $O(\sum_{l=1}^{D} M_l^2 \cdot K_l^2 \cdot C_{l-1} \cdot C_l)$, where D stands for the depth of the network, M_l denotes the length of the output feature side of the lth layer, K_l represents the size of the convolution core of the lth layer, and C_l is the number of output channels of the lth layer.

VI. Conclusion

To cope with complicated wireless signals in industrial Internet of things, an intelligent signal classification framework is developed in this paper. To extract accurate features that can fully represent the characteristics of wireless signals, the generalized envelope square spectrum and the fractional low-order cyclic spectrum are obtained through the intelligent representations of wireless signals. Then, the feature fusion neural network based on DenseNet is used to classify the signals locally on sensors receiving signals. Furthermore, the federated learning is used to fuse the learning process and recognition results of each sensor node in order to recognize the modulation recognition of signals. Finally, extensive simulation studies have been carried out to verify the effectiveness of the proposed framework. Simulation results show that the proposed framework possesses an excellent signal classification performance in industrial distributed wireless sensor networks, which significantly outperform that of the existing methods using sparse signal decomposition. In this paper, the aggregation frequency has optimization space. It can be considered to dynamically adjust the aggregation frequency according to some rules, to further reduce resource consumption without affecting performance.

References

[1] M. Liu, L. Liu, H. Song, Y. Hu, Y. Yi, and F. Gong, "Signal estimation in underlay cognitive networks for industrial internet of things," *IEEE Trans Ind. Informat.*, vol. 16, no. 8, pp. 5478-5488, Aug. 2020.

[2] M. Liu, N. Qu, J. Tang, Y. Chen, H. Song, and F. Gong, "Signal estimation in cognitive satellite networks for satellite-based industrial internet of things," *IEEE Trans Ind. Informat.*, Apr. 2020. DOI: 10.1109/TII.2020.2983390.

[3] J. Shi, M. Sha and Z. Yang, "Distributed graph routing and scheduling for industrial wireless sensor-actuator networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1669-1682, Aug. 2019.

[4] H. Kobo, A. Abu-Mahfouz and G. Hancke, "Fragmentation-based distributed control system for software-defined wireless sensor networks," *IEEE Trans Ind. Informat.*, vol. 15, no. 2, pp. 901-910, Feb. 2019.

[5] X. Feng, C. Wen, Q. Ge and X. Li, "Secure fusion filtering and clustering for distributed wireless sensor networks," in *Proc. 2019 IEEE 28th International Symposium on Industrial Electronics*, Vancouver, BC, Canada, 2019, pp. 1661-1666.

[6] P. Yildirim, K. U. Birant, V. Radevski, A. Kut and D. Birant, "Comparative analysis of ensemble learning methods for signal classification," in *Proc. 2018 26th Signal Processing and Communications Applications Conference*, Izmir, 2018, pp. 1-4.

[7] T. O'Shea, T. Roy and T. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.* , vol. 12,no. 1, pp. 168-179, Feb. 2018.

[8] T. Tuor, S. Wang, T. Salonidis, B. J. Ko and K. K. Leung, "Demo abstract: distributed machine learning at resource-limited edge nodes," in *Proc. IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops*, Honolulu, HI, 2018, pp. 1-2.

[9] Y. He, N. Zhao, and H. Yin, "Integrated networking, caching, and computing for connected vehicles: a deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 44-55, Oct. 2018.

[10] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.* , vol. 68, no. 3, pp. 3027-3032, Mar. 2019.

[11] Q. Li, N. Zhang, and X. Shen, "Channel-based optimal back-off delay control in delay-constrained industrial WSNs," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 696-711, Jan. 2020.

[12] N. Zhang, R. Wu, S. Yuan, C. Yuan, and D. Chen, "RAV: relay aided vectorized secure transmission in physical layer security for internet of things under active attacks," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8496-8506, Oct. 2019.

[13] P. Yang, F. Lyu, W. Wu, N. Zhang, L. Yu, and X. Shen, "Edge coordinated query configuration for low-latency and accurate video analytics," *IEEE Trans Ind. Informat.*, vol. 16, no. 7, pp. 4855-4864, July 2020.

[14] R. Cao, J. Cao, J. -P. Mei, C. Yin, and X. Huang, "Radar emitter identification with bispectrum and hierarchical extreme learning machine," *Multimedia Tools Appl.*, vol. 77, pp. 1-18, May 2018.

[15] X. Chen, D. Li, X. Yang, and H. Li, "Radar emitter signals identification with a optimal recurrent type 2 wavelet fuzzy neural network," *Int. J. Aeronaut. Space Sci.* , vol. 19, no. 3, pp. 685-693, Sep. 2018.

[16] X. Chen et al., "Multi-tenant cross-slice resource orchestration: a deep reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2377-2392, Oct. 2019, doi: 10.1109/JSAC.2019.2933893.

[17] J. He, P. Du and X. Chen, "Parameter estimation of communication signal in alpha-stable distribution noise environment," in *Proc. 2017 13th International Conference on Computational Intelligence and Security*, Hong Kong, 2017, pp. 182-186.

[18] G. Yang, J. Wang, G. Zhang, Q. Shao and S. Li, "Joint estimation of timing and carrier phase offsets for MSK signals in alpha-stable noise," *IEEE Communications Letters*, vol. 22, no. 1, pp. 89-92, Jan. 2018.

[19] M. Mohanty, U. Satija and B. Ramkumar, "Sparse decomposition framework for maximum likelihood classification under alpha-stable noise," in *Proc. 2015 IEEE International Conference on Electronics, Computing and Communication Technologies*, Bangalore, 2015, pp. 1-6.

[20] Z. Du, C. Wu, T. Yoshinaga, K. A. Yau, Y. Ji and J. Li, "Federated learning for vehicular internet of things: recent advances and open issues," *IEEE Open Journal of the Computer Society*, doi: 10.1109/OJCS.2020.2992630.

[21] S. H. Kong, M. Kim, L. M. Hoang, and E. Kim, "Automatic LPI radar waveform recognition using CNN," *IEEE Access*, vol. 6, pp. 4207-4219, 2018.

[22] S. Wang et al., "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205-1221, June 2019, doi: 10.1109/JSAC.2019.2904348.