Deep learning-based digital signal modulation identification under different multipath channels

Jiawen Zhang¹ | Su Hu¹ | Zhaonan Du² | Weiwei Wu¹ | Yuan Gao³ | Jiang Cao³

Abstract
Deep learning (DL) has been applied to digital signal modulation identification (DSMI) due to its powerful feature learning ability. However, most of the existing DL-based DSMI methods are limited to specific experimental scenes relating to the additive white Gaussian noise (AWGN) channel or static multipath channel. The result is that the trained network has deteriorative identification accuracy when the channel conditions change unless restrained. To solve this problem, this paper proposes a DSMI method suitable for orthogonal frequency division multiplexing (OFDM) under different multipath channels, including the variation of delay, path number and channel coefficient. This method can accurately detect the modulation feature rather than the channel's to identify the modulation type, thus reducing the network training amount. The method is divided into two parts. Firstly, traditional signal processing methods are combined, including various channel estimators and equalisers to compensate for the channel. Then a robust DL network, RSN-MI, is designed as a classifier. Unlike other DL-based DSMI methods, the influence of signal processing algorithms on DSMI performance are focused on rather than model parameters. Besides, the proposed classifier is compared with the DSMI classifier in other contributions. The results show that the classifier works better in different multipath channels.

1 | INTRODUCTION
Digital signal modulation identification (DSMI) is of great significance in wireless communication systems. As a necessary step between signal detection and demodulation, it has attracted more and more attention. The main task of DSMI is to identify the modulation type of the received signal without using any prior knowledge or using a small amount of prior knowledge, which lays the foundation for subsequent signal demodulation and information acquisition. DSMI plays an important role in both military and civil fields [1], especially in military fields, modulation identification is the prerequisite of jamming and monitoring enemy communications. It can also be used for spectrum detection and signal confirmation etc. in civil fields.

Two traditional digital signal modulation identification approaches have been extensively studied: one is the maximum likelihood (ML) approach, and the other is the pattern recognition (PR) approach [2]. The maximum likelihood approach is based on the likelihood function of received signal, and the likelihood ratio is compared with the appropriate threshold value to achieve the best classification effect under the minimum error criterion of Bayes [3–6]. Due to the need for more prior knowledge and high algorithm complexity, this approach’s application has great limitations. For the pattern recognition approach, it is necessary to transform the received signal to extract features in different dimensions. Higher-order cumulants (HOC), wavelet transform, and signal power ratio are often selected as feature parameters. Then a suitable machine learning classifier, such as K-nearest neighbour (KNN), Decision Tree (DT) etc., is used to identify the modulation type [7–10]. The identification performance of this approach is affected by feature selection, and it is difficult to extract the same feature from multiple modulation types. Besides, when the task data volume is large, this approach is not efficient.

With the development of computer science and hardware, Deep learning (DL) technology has been developed rapidly. Deep neural network has been widely applied in classification and object detection tasks which has shown excellent performance. In recent years, considering the massive data generated by the wireless communication system, deep learning has been gradually employed to DSMI tasks and has demonstrated its strong feature learning ability. In [11, 12], digital...
signals are converted into constellation diagrams as data sources, and then two well-known convolutional neural network (CNN) models, AlexNet and GoogleNet, are used for model training to identify ASK, PSK, and QAM signals successfully. An IC-AMCNET network model is proposed in [13], which can be applied to communication scenarios with deficient delay requirements. This model can greatly reduce the computation complexity. In [14], the cyclic spectrum is first adopted to preprocess the signal, and then the deep autoencoder network is used for modulation identification. The authors from [15] propose a novel deep hierarchical network that can combine high-level and low-level feature maps to predict the modulation type. In [16], a multi-modality fusion model is adopted to incorporate the handcrafted features and image features extracted from received signals. The identification accuracy gets a further improvement with the joint features.

However, these methods focus on the impact of neural network structure and parameters on the classification effect. All of the methods mentioned above have common limitations that trained model’s performance will deteriorate when the channel environment changes. Most of the work related DSMI is obtained in the additive white Gaussian noise (AWGN) channel, or although the data is received in a multipath environment, there is no communication-related processing on the signal. The result is that a trained network cannot effectively identify the modulation type of the signal which is generated from other channel environments, so it cannot be used in actual wireless communication systems. Besides, almost all of the current communication systems are built based on OFDM technology, but there are few OFDM related DSMI studies [17–19]. Therefore, it is of great research value to find an effective DSMI method suitable for multicarrier OFDM communication systems.

Considering the above problems, this paper proposes a DSMI method suitable for multicarrier OFDM systems under different multipath channels, including the variation of delay path number, and channel coefficient. The method proposed in this paper is carried out in two steps. In the preprocessing stage, we use a second-order statistical blind (SOSB) channel estimator and a subspace blind (SB) channel estimator to obtain the fading coefficient of multipath channels. The purpose of using the blind estimation algorithms are to apply our method to non-cooperative communication. They are also compared with the least square (LS) estimator and minimum mean square error (MMSE) estimator based on the known sequence. Then the zero-forcing (ZF) equaliser and MMSE equaliser are used to recover the received data from different channel environments. In the identification stage, we propose a deep neural network model RSN-MI suitable for DSMI tasks, which is an improvement on the deep residual shrinkage network (DRSN) model. RSN-MI adopts the modified soft threshold update block as the nonlinear layer to reduce the influence of high noise on the feature extraction of the digital signal. In order to build up the optimal classifier for DSMI, we consider the features of different modulation types to design our network. Compared with other DSMI classifiers, the advantage of RSN-MI is that it can detect the digital signal waveform feature at different modulation types and avoid the influence of the channel feature on network training.

Finally, we evaluate the performance of all different signal preprocessing methods adopted in this paper on DSMI, which are critical to realise the correct identification of received signals in different channel environments. Besides, we compare the channel-varying adaptability of our network model with other DSMI methods from [6], [7], and [18] include ML-based average likelihood ratio test (ALRT), two PR methods with HOC, and a CNN DL model. Compared with both DL-based DSMI methods or traditional’s, our method does not need to train the model for each fixed multipath transmission scene, but it can be applied to the different multipath channel environment after training once or in a small amount. In other words, it is a robust digital signal modulation identification method. We also analyse the computation complexity of these methods mentioned in this paper, which further illustrate our work’s feasibility.

In this paper, the following notations are used: $x$, $X$ represent a scalar, a vector, and a matrix, respectively. The superscript $[^t]$ is the operation transpose, Hermitian and conjugation. $E\{\cdot\}$ calculates the mathematical expectation and $||\cdot||$ defines the Frobenius norm.

The rest of the paper is arranged as follows. Section 2 introduces the OFDM system model; The signal preprocessing methods are introduced in Section 3, including various channel estimation algorithms and equalization algorithms; Section 4 introduces the proposed network model RSN-MI; Section 5 is the data sampling platform built for acquiring the datasets in the real channel environment; Section 6 is performance evaluation and comparison with other methods; Computation complexity of these methods mentioned in this paper are analysed in Section 7; Section 8 concludes our work.

2 | OFDM SYSTEM MODEL

Figure 1 shows the OFDM transmission model used in this paper. $\tilde{s}_N(k) = [\tilde{s}_1(k) \tilde{s}_2(k) \ldots \tilde{s}_N(k)]^T$ is the $k$th digital signal vector after bit mapping, where $N$ represents the number of digital symbols carried on each OFDM symbol, that is, the number of sub-carriers. The OFDM symbol $s_N(k) = [s_1(k) s_2(k) \ldots s_N(k)]^T$ in the time domain can be expressed as the inverse discrete Fourier Transform (IDFT) of $\tilde{s}_N(k)$ as $s_N(k) = F_{N}^{T} \tilde{s}_N(k)$ where $F$ is the DFT matrix and $F_{N}^{T}$ the DFT matrix. Let $s_{cp}(k) = [s_{N}^{(1)}(k) s_{N}^{(2)}(k) \ldots s_{N}^{(P)}(k)]^T$ represents the $k$th OFDM symbol after adding the cyclic prefix (CP). This operation can be achieved by a matrix $T = [I_{cp} \quad I_{N}]$, where $I_{cp}$ is the matrix formed by the last $D$ rows of $I_{N}$. The usually selected number of CP points $D$ is equal to $1/4N$, and $P = N + D = 5D$.

The multipath channel can be simulated by a tap delay line (TDL) model [20], and its discrete impulse response is expressed as follows:

$$c(n) = \sum_{j=0}^{L-1} h_j \delta(n - \tau_j)$$  \hspace{1cm} (1)
where $L$ represents the number of paths, and it satisfies $L \leq D$. $b_l$ represents the channel fading coefficient of the $l$th channel, and its amplitude obeys the Rayleigh distribution. $\tau_l$ is the path delay normalised by the sampling time $T_s$ of the $l$th path, written in vector form as $h = \{b_0, b_1, ..., b_{L-1}\}^T$.

The signal at the receiving end can be convolved to simulate the linear superposition of signals from different paths. The inter-symbol interference caused by the multipath effect causes the received $k$th OFDM symbol $r_c(k)$ to be generated jointly by $s_c(k)$ and $s_c(k-1)$, which can be represented as follows:

$$r_c(k) = \begin{bmatrix} H_1 \\ H_0 \end{bmatrix} \begin{bmatrix} s_c(k-1) \\ s_c(k) \end{bmatrix} + w_p(k)$$  \hspace{1cm} (2)

where $H_0$ is the $P \times P$ topelise matrix composed of 0 elements and the channel coefficient with first row $[b_0 \ 0 \ ... \ 0]^T$ and first column $[b_0 \ ... \ b_{L-1} \ 0 \ ... \ 0]^T$. $H_1$ is the $P \times P$ topelise matrix with first row $[0 \ ... \ 0 \ b_{L-1} \ ... \ b_1]^T$ and first column $[0 \ 0 \ ... \ 0]^T$. The white Gaussian noise added to each symbol can be expressed as $w_p(k) = [w_1(k), w_2(k), ..., w_P(k)]^T$. $\tilde{r}_N(k) = [r_1(k), r_2(k), ..., r_N(k)]^T$ can be obtained after using $R = [0_{cp}, I_N]$ to remove CP of $r_c(k)$.

$\tilde{r}_N(k) = [r_1(k), r_2(k), ..., r_N(k)]^T$ is the frequency modulation symbol block obtained through DFT.

Assuming that all sub-carriers of the OFDM symbol use the same modulation type, the purpose of this paper is to correctly identify the modulation type of the OFDM symbols within $C = \{\text{BPSK, QPSK, 8PSK, 16QAM, 32QAM, 64QAM}\}$. However, due to the influence of multipath channels, the received symbols will be distorted and stretched. The distribution of $\tilde{r}_N(k)$ in the 2D space is shown in Figure 2.

At present, most of the literature about the modulation identification of digital signals based on deep learning ignores this problem, which is obviously unreasonable. It is difficult for deep neural networks to extract modulation features from distorted digital symbols interfered with by multipath wireless channels.

### 3 SIGNAL PREPROCESSING

This section mainly introduces the digital signal preprocessing algorithms used in this paper, including various channel estimators and channel equalisers, whose purpose is to eliminate the
interference of multipath effect to the data sequence, which is the key to realise the robustness of our method. We focus on evaluating the influence of different signal preprocessing algorithms on DSIMI, and this result is shown in Section 6.

3.1 | SOSB channel estimator

The second-order statistics blind channel estimation algorithm is to estimate the autocorrelation matrix $R_r$ of the received signal and recover the channel coefficient vector $h$. The algorithm does not use or uses little prior knowledge to obtain a more accurate estimate, so it is very suitable for DSIMI tasks [21]. Let $\tilde{H}_0$ and $\tilde{H}_1$ be the Toeplitz matrix with the size of $D \times D$ in the upper left corner and upper right corner of $H_0$ and $H_1$, respectively. Then Equation (2) can be rewritten in Equation (3) which is shown at the below. In (3), $s_N^k(k)$ is a vector composed of $(i-1)D+1$ to $iD$ elements of $s_N(k)$, $1 \leq i \leq 4$.

$$r_{cp}(k) = \begin{bmatrix} \tilde{H}_1 & 0 & 0 & \tilde{H}_0 \\ 0 & \tilde{H}_0 & 0 & \tilde{H}_1 \\ 0 & 0 & \tilde{H}_1 & \tilde{H}_0 \\ 0 & 0 & 0 & \tilde{H}_0 \end{bmatrix} [s_N^1(k-1), s_N^1(k), s_N^2(k), s_N^2(k)]^T + w_p(k)$$

When the transmission symbols are independent of each other and the average value is 0, the autocorrelation matrix $R_r = E[r_{cp}(k)r_{cp}^H(k)]$ of the received signal can be obtained, which has the following structure:

$$R_r = \begin{bmatrix} \sum_{k=0}^{D-1} \tilde{H}_{0,0} & \cdots & \tilde{H}_{0,D-1} \\ \vdots & \ddots & \vdots \\ \tilde{H}_{D-1,0} & \cdots & \sum_{k=0}^{D-1} \tilde{H}_{D-1,D-1} \end{bmatrix} + \sigma_r^2 I$$

where $\tilde{H}_{0,0} = \tilde{H}_0 \tilde{H}_0^T$, the element in the first column is $h_0^T [h_1 \ldots h_{D-1}]$. $\sigma_r^2$ is the variance of the noise. $R_r$ can be replaced by $\hat{R}_r = \frac{1}{K} \sum_{k=1}^{K} r_{cp}(k)r_{cp}^H(k)$. Take the last $D$ elements in the first column of the $\hat{R}_r$ to get the channel coefficient which contains the fuzzy factor $b_0$. $b_0$ can be estimated by little prior knowledge, which is called the semi-SOSB algorithm.

3.2 | SB channel estimator

Based on the orthogonality of the received signal's correlation matrix and its noise subspace $E_N$, the channel coefficient vector $h$ can be estimated by the mathematical relationship between transmission blocks. The blind subspace estimation algorithm is generally used in the case of $D = 1/4N$ without changing the transmitter's structure [22]. Define the vector:

$$\bar{r}_{cp}(k) = [r_{cp}^1(k-1)^T \ldots r_{cp}^4(k-1)^T r_{cp}^2(k)^T \ldots r_{cp}^4(k)^T]^T$$

$$\bar{t}_{cp}(k) = [t_{cp}^1(k-1)^T \ldots t_{cp}^4(k-1)^T t_{cp}^2(k)^T \ldots t_{cp}^4(k)^T]^T$$

$$\bar{w}_{cp}(k) = [w_p^1(k-1)^T \ldots w_p^4(k-1)^T w_p^2(k)^T \ldots w_p^4(k)^T]^T$$

From Equation (2), we can get $\bar{r}_{cp}(k) = H\bar{t}_{cp}(k) + \bar{w}_{cp}(k)$, where $H$ is the matrix of size $(2N + D) \times 2N$ defined in [23]. Define $R_{rr}$ and $R_{ss}$ to be the autocorrelation matrix of $r_{cp}(k)$ and $s_{cp}(k)$ respectively. Then $r_{cp}(k)$ can be represented as follows:

$$R_{rr} = H(h)R_{ss}H(h)^H + \sigma_r^2 I_{2N+D}$$

when $R_{ss}$ is full rank, the following cost function can be constructed:

$$J(h) = \min h^H Qh \quad \text{s.t. } ||h|| = 1$$

where $Q$ is a matrix of size $(D + 1) \times (D + 1)$, which can be constructed by the normalised eigenvector corresponding to the $D$ smallest eigenvalues of $R_{rr}$. The SB estimation algorithm also has a fuzzy factor, which can be eliminated by the semi-SB algorithm through a few pilots.

3.3 | Pilot-based estimators

Two pilot-based estimators, LS estimator and MMSE estimator are adopted to compare with the blind estimators. LS estimation algorithm [24] constructs the cost function according to the least square criterion and obtains its minimum value through gradient operation.

$$\hat{H}_{LS} = X^{-1}Y$$

where $X$ and $Y$ are the frequency domain pilot sent and received respectively. $\hat{H}_{LS}$ is the frequency domain response of the estimated channel.

The MMSE estimation algorithm obtains the channel matrix $\check{H}_{MMSE}$ in a statistical sense by minimising the mean square error of the actual transmitted pilot $X$ and the estimated $\check{X}$. In actual application, The LMMSE algorithm is usually used instead, as shown below:

$$\hat{H}_{MMSE} = R_{HH} \left( R_{HH} + \frac{\hat{\beta}}{SNR} I \right)^{-1} \hat{H}_{LS}$$

where $R_{HH}$ is the autocorrelation matrix of the channel, $\hat{\beta}$ is a constant related to the modulation type, and $SNR$ is the defined average SNR in [25].
### 3.4 Channel equaliser

ZF equaliser and MMSE equaliser are used to compensate for channels. ZF equaliser is often used for signal preprocessing in various wireless communication systems due to its simple implementation. The realisation algorithm is expressed as:

\[
\hat{L} = \begin{bmatrix} Y_1 & Y_2 & \cdots & Y_N \end{bmatrix} \begin{bmatrix} H_1 & H_2 & \cdots & H_N \end{bmatrix}^T \tag{10}
\]

where \(Y_i\) and \(H_i\) are the received signal and channel response on the \(i^{th}\) subcarrier in the frequency domain after DFT respectively. \(\hat{L}\) represents the estimated signal sequence.

MMSE equaliser is also based on the minimum mean square error criterion, and the received signal can be estimated as follows:

\[
\hat{L} = \hat{H}^H \left( \hat{H} \hat{H}^H + \gamma^{-1} I_N \right)^{-1} Y \tag{11}
\]

where \(\hat{H}\) is the channel matrix obtained by the estimator, represents the SNR of the communication chain, and \(I_N\) is the identity matrix of size \(N \times N\).

### 4 RSN-MI

CNN is the most popular deep learning framework at present. Its most important component unit is a filter using convolution operation, which is usually called convolution kernel. Because the convolution operation can describe image features well, it has been widely used in image recognition tasks. At present, much literature in the field of communication adopts the traditional CNN framework to deal with DSMI tasks, and IQ sampling is directly used for model training, which lacks reasonable interpretation. The typical structure of CNN is difficult to extract the features of different modulation types from the data sequence, and the extracted features are seriously disturbed by the multipath environment.

#### 4.1 Dynamic soft threshold update block

In this section, we fully consider the features of wireless communication transmission signals and propose an RSN-MI network structure. This network structure is based on the deep residual shrink network (DRSN), modified to be suitable for DSMI tasks. DRSN is a recently proposed deep learning network for mechanical fault diagnosis [26]. This structure consists of a series of residual shrink building units (RSBUs). In addition to the traditional CNN elements in RSBU, such as the convolutional layer, activation function, and batch normalisation (BN), the most important is introducing a dynamic threshold update block as shown in Figure 3, which makes our network have excellent identification accuracy in a high-level noise environment.

The dynamic soft threshold update block comprises a soft threshold function (i.e. a popular shrinkage function) [27] and a dynamic threshold generator. The threshold function can effectively remove the redundant noise features in the feature map, meanwhile retaining more digital signal modulation features. Essentially, the soft threshold function is equivalent to a new activation function, expressed as follows:

\[
y = \begin{cases} 
  x - \tau & x > \tau \\
  0 & -\tau \leq x \leq \tau \\
  x + \tau & x < -\tau 
\end{cases} \tag{12}
\]

where \(x\) is the feature value at the corresponding position on the feature map, \(\tau\) is the estimated soft threshold, \(y\) is the modified feature value.

The soft threshold function is inserted into RSN-MI as a nonlinear layer to update the feature map, setting the smaller
feature value (close to 0) to 0. These inputs have a high probability of being noise without useful information. Besides, for the signal data set, when the bitstream is mapped to the modulation constellation, the value of the IQ sequence can be positive or negative, which is not like picture pixels, only positive values. For the DSMI task, it is necessary to preserve the signal’s negative features as much as possible. The threshold function structure can guarantee this very well, unlike ReLU who loses useful information for data training by transforming negative inputs into zero. But its gradient has the same form as ReLU, which ensures that the parameters can be effectively trained.

The dynamic threshold generator is achieved through a DL-based attention mechanism. Different from the traditional fixed threshold update block, manual experience is required. Threshold dynamic update can ensure that the threshold of the threshold function is automatically updated, reducing the complexity of the threshold setting and improving the identification accuracy. More specially, the attention mechanism is a brand-new feature recalibration strategy. By modeling the correlation between feature channels, the weight coefficient \( \alpha \) is automatically updated, reducing the complexity of the threshold setting and improving the identification accuracy. More specifically, the attention mechanism is a brand-new feature recalibration strategy. By modeling the correlation between feature channels, the weight coefficient \( \alpha \) is automatically updated, reducing the complexity of the threshold setting and improving the identification accuracy.

In RSN-MI, the dynamic threshold generator draws on the weight coefficient’s learning mechanism in squeeze-and-excitation networks and uses the coefficient to dynamically generate soft thresholds by weighting the feature map. As shown in Figure 3, The weight coefficient learning is realized by combining the FFD block (flattern operation, fully connected (FC) layer and dropout) and two FC layers. First, the FFD block performs a “squeeze” operation on the feature map with input size \((H, W, C)\) and outputs feature data with size \((1, 1, C)\). Then two FC layers with the same number of neurons as channel’s are used to obtain each channel’s corresponding weight coefficient. The sigmoid function ensures that the coefficient range is reduced to \((0, 1)\). This process is called “excitation”.

After obtaining the weight coefficient \( \alpha \), \( 1 \leq i \leq C \), the soft threshold corresponding to each channel can be generated according to the following equation:

\[
\tau_i = \alpha_i \cdot \| \text{FFD}(X) \|, \quad 1 \leq i \leq C
\]

where \( \tau_i \) represents the \( i \)th channel’s soft threshold, \( \| \text{FFD}(X) \| \) represents the result of processing feature map \( X \) by using the FFD block, \( \| \cdot \| \) is the absolute value operator, whose purpose is to ensure that the resulting threshold is positive. The dynamic soft threshold obtained in this way can be adjusted adaptively according to the feature input, and it can be compressed to a small value by weighting \( \alpha_i \) to meet the needs.

### 4.2 Network structure

With the help of dynamic soft threshold updating structure, RSN-MI is proposed and applied to DSMI task. The proposed RSN-MI is constructed according to the features of the IQ data sequence and the size of the selected modulation types, which has strong interpretability, as shown in Figure 4. \( \forall \chi \subset \mathbb{C}^2, \chi \in \mathbb{C}^2 \) and \( |\chi| = M \) where \( M \) is the size of the selected modulation types. For \( \forall \chi \) it can be regarded as a point distribution in a 2D space. Therefore, we can integrate the constellation features of each modulation type into the network construction.

Our network’s input is an IQ sampling sequence \( \chi \) with the size of \((2, 2400)\) after OFDM demodulation. The length of 2400 is chosen because the distribution of more signal sampling points will not be changed by noise, where the network can analyze more dimensional information of the data. The convolutional layer is used as the first two layers of the network with kernel sizes \((1, 2)\) and \((1, 4)\) whose primary purpose is to extract the single-dimensional features of the IQ sequence. In detail, a small kernel of size \((1, 2)\) is used to accurately extract the feature range to BPSK signals with only two mapping points. The maximum feature extraction range can be expanded to 64QAM by combining the kernel with size \((1, 4)\) because there are at most eight values in each dimension of I and Q when \( \chi \) adopts 64QAM. All these two kernels can extract all the high and low-order modulation features. Additionally, the number of channels per layer is set to 50.

The feature map, which is generated due to the work of the first two convolutional layers, is sent into the modified RSBU. As the core unit of our network, RSBU can reduce the interference of noise to model training. It improves the feature learning ability of RSN-MI in a high-noise environment. In this unit, the BN layer is adopted to modify the data distribution and improves the training speed. Considering that the value distribution of the digital signal is different from the image pixel value, the features are positive and negative. So, we use the parametric rectified linear unit (PReLU) function instead of the Relu function to increase the nonlinearity of our model while retaining the negative feature of the signal as much as possible. Then a convolutional layer with a kernel of \((2, 1)\) and several channels of 50 is used to extract richer features of data combined the two dimensions of I and Q. It is worth mentioning that the 2D convolution kernels are adopted in RSN-MI rather than 1D convolution kernels, because using a 1D kernel will lose one dimension of data information. For the DSMI task, the 1D CNN may lose the phase features of the modulated signal, which leads to a worse identification performance. Besides, our modified RSBU is set according to the receptive field of the 2D convolution kernel. These factors determine a more flexible 2D convolution is adopted in RSN-MI.

After that, the soft threshold update block starts to work. As described in Section 4.1, the soft threshold update block adaptively adjusts the threshold according to the input feature map. Subsequently, the generated threshold is sent into the soft threshold nonlinear layer, which updates each channel of the feature map according to the rules set by Equation (12). This layer can effectively eliminate noise features by setting near-zero inputs to zero. Then, cross-layer training can be achieved with the “identify shortcut” connection. The “identify shortcut” connection can quickly propagate the deeper layer’s gradient back to the shallow
layer, preventing poor parameter training due to gradient dispersion, which improves training speed and model robustness.

Finally, the FC layer with 6 neurons is adopted, and the softmax function is used to convert the input samples into corresponding identification probability. The softmax function is expressed as follows:

$$S_i = \frac{e^{V_i}}{\sum_j e^{V_j}}$$

where $V_i$ represents the output of the $i$th neuron. $S_i$ is the probability that the test data belongs to the $i$th modulation type. Because of the use of soft thresholds, RSN-MI can well eliminate feature values close to 0, which are generally associated with noise, and all convolutional layer settings take into account the characteristics of data sequence. It works better in DSMI tasks than traditional convolutional neural networks, all of which we will show in Section 6.

4.3 | Experiment platform

The deep neural network in this paper is built based on Keras 2.3.1 running on top of Tensorflow 2.2.0. Tensorflow is a symbolic mathematical system based on dataflow programming, which is widely applied to the programming implementation of various machine learning algorithms. All of our results are run on four parallel NVIDIA GeForce GTX1080Ti GPU and Intel(R) Xeon(R) Silver 4110 CPU @ 2.10 GHZ.

5 | DATA SAMPLING PLATFORM

In order to improve the reliability of the experiment, we built an OFDM-based data sampling platform in a static indoor office environment, as shown in Figure 5. The IQ sequences used in this paper, including the training set and test set, are all collected from the Universal Software Radio Peripherals (USRPs). In our platform, two USRP-2943Rs are used as software defined radio (SDR) nodes. One USRP-2943R is used as a transmitter, of which RF0 is enabled with active antenna Tx1. The other one is used as a receiver, and the antenna Rx2 of the RF1 channel is activated to receive the signal from the air interface. The receiver trigger sampling mode is set as “Tx start trigger” to ensure that valid data can be collected.

In addition to the SDR part, the transmitting and receiving processes based on OFDM signals are set up in LabVIEW. LabVIEW is a graphical programming language development environment, which can easily interact with USRP to collect data, and it can also embed the MATLAB script module for mixed programming to improve work efficiency. In the
transmitter part, the OFDM signal frame based on 802.11a is generated with different modulation types, where 64 subcarriers are used to carry data, and the length of CP points is 16. The short training sequence (STS) and the long training sequence (LTS) are inserted in the frame header, which is used to estimate the sample timing offset (STO) and the carrier frequency offset (CFO).

Formula (1) is programmed as a multi-fading module to be embedded into LabVIEW before signal transmission to improve the channel diversity in the experimental environment, where the delay, multipath number and channel coefficient of the multi-fading module are all dynamic changes. These parameters are specifically set by the 802.11a working group designed to predict modulation in a real office environment [28]. These physical configurations result in the signal transmitted by each SDR node being in a different multipath channel. When the data frame is generated, it is sent to the air by the activated antenna, where the centre frequency of the carrier is 2.35 GHz, and the effective bandwidth of the signal is 20 MHz. In the receiver part, the data collected by the SDR node is synchronised and power normalised, and then the processed data is sent to MATLAB for various signal processing described in the third part to obtain the test set. It should be noted that when collecting data to generate a training set, the multi-fading module is not used to change the channel. The purpose of this is to minimise the interference of multipath features to network training. In this case, the received OFDM signal frame is synchronised and power normalised, then these data after FFT operation constitutes the training dataset.

Following the identification method used in this paper, we generate three datasets with the data Sampling platform. We assume that all OFDM symbols adopt the same modulation type, which can be selected from BPSK, QPSK, 8PSK, 16QAM, 32QAM, and 64QAM. The first dataset is used for DL-based DSMI methods, which works with RSN-MI proposed in this paper and the CNN-based classifier proposed in [18], consisting of the train set and test set. The train set consists of $5 \times 10^4$ samples collected for each modulation type, and the total number of samples is $3 \times 10^5$, in which the ratio of the training set and verification set is 8:2. The test set comprises various preprocessed IQ sequences. To collect test sequences under different channel conditions, our multi-fading module exerts an effect on the transmitting signal, which produces a variable multipath channel. The test set's size is $10^4$ samples for each modulation type, and the total number of samples is $6 \times 10^4$ for each preprocessing method. Unlike other works, each test sample’s channel parameters are randomly generated, including the delay, path number, channel coefficient to show the robustness of our methods.

The second dataset operates on the two PR methods. Based on the first dataset, the high order cumulants are further extracted as signal features, then forming the new dataset. The third dataset comprises phases of the IQ sequence in dataset one because the likelihood function is created with the phase as an unknown parameter.

## 6 PERFORMANCE EVALUATION

This section shows the performance of our method through specific experiments. Besides, we compare the channel-varying adaptability of our network model with other DSMI methods including ML-based ALRT, two PR methods with HOG, and a CNN deep learning model. All the results are based on the three datasets described in Section 5.

### 6.1 Impact of signal preprocessing on DSMI

In this experiment, we compare the influence of all different signal preprocessing methods adopted in this paper on DSMI which is shown in Figure 6. This is a key step to realise the correct identification of received signals in different channel environments. All results are run based on RSN-MI, indicating that our network is robust to various signal processing algorithms.

For the channel estimators based on the pilot, as shown in Figure 6(a), Using the same equaliser, the average identification accuracy associated with the LMMSE estimator is better than that of the LS estimator, especially in the case of low SNR. At SNR $= 10$ dB, the difference between these two estimators reaches a maximum of about 5%, which is reduced with the increase of SNR. Besides, we can also clearly see that with the same estimator, the results corresponding to the MME equaliser is more superior when compared to that of the ZF equaliser at low SNR, because the MME equaliser takes into account the statistical characteristics of the signal. Therefore, its performance is less affected by noise. Figure 6(b) shows the performance of two semi-blind estimators in DSMI. All the curves show an upward trend as the SNR increases. Compared to the semi-SOSB estimator, the semi-SB estimator results in a significant improvement at all SNR, which can reach 100% at SNR $= 24$ dB with the same equaliser. It can also be seen that MMSE equaliser performance is better than ZF equaliser at all SNR.

The performance of all the adopted channel estimation algorithms is evaluated in Figure 6(c) with MMSE equaliser. At SNR $= 10$ dB, the identification accuracy of all algorithms can reach more than 90%, which shows our method is robust. In addition, the result of the semi-SB estimator has little difference from that of the MMSE estimator and their identification accuracy can reach 100% with the increase of SNR. This shows that the semi-SB blind estimator can estimate the channel coefficient as accurately as the pilot-based estimators, and the effect is even better in a certain SNR interval. In addition, The accuracy associated with the semi-SOSB estimator can also reach approximately 98% at SNR $= 24$ dB. This shows that our method can be extended to non-cooperative communication. Figure 6(d) shows the recognition accuracy for all six adopted modulation types corresponding to the semi-SOSB estimator and MMSE equaliser. We can see from the results that PSK signals are easier to identify than QAM signals. At SNR $= 10$ dB, BPSK, QPSK and 8PSK can be identified by almost 97% while it is difficult to
identify the QAM signal, especially the 64QAM which is represented by the bottom curve drawn in Figure 6(d). Finally, with the increase of SNR, the identification performance of all modulation types can almost reach 1.

### 6.2 The proposed RSN-MI versus other DSMI classifier

To further demonstrate the better adaptability of our proposed methods to channel environment changes, we compare the identification performance for signal modulation types under different multipath channels, which run on the RSN-MI classifier proposed in this paper and other DSMI classifiers, including DL-based methods and traditional’s. For the ML method, we take the phase of the signal as the unknown parameter to establish a likelihood function, then the modulation type is determined by the ALRT criterion. In the PR test, based on the higher-order cumulant extracted from the received signal, KNN and DT are selected as classifiers to identify the modulation type. Besides, the CNN-based classifier proposed in [18], called CNN-MI, is also used to compare with our deep learning model. With the same hardware condition and the datasets in

![Figure 6](image_url)

**FIGURE 6** Identification accuracy vs. SNR using different channels estimators and equalisers with our RSN-MI network. (a) Average accuracy with the digital signal that had undergone pilot-based estimators, (b) average accuracy with the digital signal that had undergone semi-blind estimators, (c) average accuracy with the digital signal that had undergone all channels estimators in this paper and MMSE equaliser. (d) Identification accuracy of six modulation types with the digital signal that had undergone semi-SOSO estimator and MMSE equaliser

![Figure 7](image_url)

**FIGURE 7** Modulation identification accuracy compared with other DSMI classifiers based on the semi-SB estimator and the MMSE equaliser

Section 5, the corresponding result in Figure 7 is obtained. The first thing to emphasise is that all curves are drawn using the signals received in different multipath channels preprocessed by the semi-SB estimator and the MMSE equaliser.
As shown in Figure 7, it is clear that the highest curve comes from our network model, whose performance is far ahead of the other classifiers selected in this paper. The result of RSN-MI is close to 98% at SNR = 10 dB. With SNR increasing, it can reach 100% eventually. It shows that our network model can work well at low SNR, and it is robust working on the test data from different multipath channels. In contrast, the performance of other classifiers is much less. For CNN-MI, another deep learning-based DSMI model, its performance at low SNR is less than 60%. This result shows that when the channel environment changes, the general CNN-based classifier can no longer identify the modulation type of the received signal, although the multipath effect has been compensated through the signal processing algorithm.

In terms of the ML method, the ALRT-based classifier has good classification performance at low SNR, which is over 70%, but with the increase of SNR, there is no significant improvement. This is due to the approximate probability density function selected for phase parameter deviating from the actual value at high SNR. The model mismatch results in poor performance. Compared with PR methods denoted as “KNN+HOC” and “DT+HOC” respectively. When SNR = 10 dB, they have a poor classification accuracy, only 30%, which indicates that these methods cannot be used in the multipath channel with variable parameters. Their performance is inferior to other DSMI methods before SNR is less than 18 dB, for the feature selection and simple machine learning classifier structure. Through the above analysis, RSN-MI can better learn the modulation feature rather than the channel feature so that better performance can be achieved.

Figures 8 and 9 show the confusion matrices evaluation of CNN-MI and RSN-MI at SNR = 10 dB with two semi-blind estimators and the MMSE equaliser, respectively. It is clear from Figure 8 that the matrices are not in complete diagonal form. The results in Figure 8(a) shows that only BPSK and QPSK can be distinguished correctly when CNN-MI is used as the classifier with the semi-SOSB estimator while 16QAM, 32QAM and 64QAM are all incorrectly identified as 8PSK. Although this result has improved when the received signal is preprocessing by the semi-SB estimator as shown in Figure 8(b), it still has a big performance gap compared with our proposed RSN-MI as a classifier. As shown in Figure 9, both confusion matrices are in diagonal form. Regardless of whether the semi-SOSB estimator or the semi-SB estimator is used, RSN-MI has accurate identification capabilities which can clearly identify the modulation type of the signal from the fading channels with different parameter settings when SNR is low. This comparison result shows that our method applies different channel environments and has better performance advantages.

### 7 COMPUTATION COMPLEXITY OF DSMI METHODS

In this section, we analyse the computation complexity of all DSMI methods used in this paper, including the proposed RSN-MI and another DL model, CNN-MI, as well as two traditional DSMI methods, which will help to evaluate the practical application value of these algorithms.

For two DL models, operational intensity (OI) of roof-line model theory is adopted to quantitatively analyse their deployment capability on the hardware platform. OI is defined as time complexity (TC) divided by space complexity (SC), where floating-point operations (FLOPs) are used to evaluate the TC and the total memory consumption to the SC. Thus, it indicates how many FLOPs per byte of memory can be used by the DL model during training. Since the blocks of RSN-MI generating computation are mainly the first three convolutional layers and the three FC layers in RSBU, while the complexity of the other blocks can be almost ignored, the author provides a feasible
the running time of all DSMI methods in practical project. Relying on the same hardware platform described in Section 4.3, we estimate the running time of these algorithms in actual tasks, including pre-processing time (PT) using the semi-SB estimator and MMSE equaliser, feature extraction time (FET) with HOC, training time (TRT) and test time (TET). As shown in Table 2 (each step not taken by a particular algorithm is replaced with a "/"), the PT of the signal arriving at the receiving end is more prolonged than the FET, but the pre-processing is needed for every DSMI method, while feature extraction is only used for the method based on PR, increasing the extra time cost about 31 ms. Besides, compared with traditional algorithms, the DL model requires a

| DSMI methods | PT (s/sample) | FET (s/sample) | TRT (s/model) | TET (s/sample) |
|--------------|--------------|---------------|--------------|---------------|
| RSN-MI       | 4.12 × 10^{-2} | /             | 496          | 1.63 × 10^{-4} |
| CNN-MI       | 4.12 × 10^{-2} | /             | 1094         | 5.82 × 10^{-4} |
| ALRT         | 4.12 × 10^{-2} | /             | /            | 3.18 × 10^{-1} |
| KNN+HOC      | 4.12 × 10^{-2} | 3.11 × 10^{-4} | 5.85 × 10^{-1} | 8.00 × 10^{-6} |
| DT+HOC       | 4.12 × 10^{-2} | 3.11 × 10^{-4} | 3.63         | 3.55 × 10^{-5} |
longer training time, and the training time of CNN-MI is more than twice that of RSN-MI. But the DL model can work well once the training is completed, without repeated training. In practical application, for a sample classification test, the ML-based ALRT algorithm consumes much more time than other's, about 3.18×10⁻¹ s, which determines it can’t be used normally in practical tasks due to a considerable time delay. For other algorithms, the actual time consumed by the DL model is “PT+TET”, while the PR methods about “PT+FET+TET”. Although the TET of the PR methods is smaller than that of the DL-based methods, the DL model has the lowest total test consumption time, where RSN-MI is the best with 4.1363×10⁻² s. Therefore, our method has high practical application value.

8 | CONCLUSION

In this paper, we propose a DL-based DSMI method suitable for multicarrier OFDM systems under different multipath channels, including the variety of delay, path number, and channel coefficient. Our method is divided into two steps: Firstly, different channel estimators and equalisers are used to preprocess the signal; Then, we improve a new DL model based on DRSN as a modulation classifier to make it suitable for DSMI tasks. Compared with other DSMI methods, whether the DL-based methods or the traditional’s, our model can avoid channel features and extract unique features of modulation types under a high noise environment. In the experiment, we mainly evaluate the performance of various signal preprocessing methods on DSMI. The experiment results indicate that our method can be applied to different multipath channels with only a small amount of model training. Besides, when considering the qualitative criteria, our method’s prediction performance is much better than other DSMI methods. Finally, the complexity analysis shows that our model can make decisions within a tolerable time delay.

Based on our research, there are still some points worth considering about DSMI. In future, we will consider how to implement our method in completely non-cooperative communication, which involves the research of blind signal processing algorithms. In a multuser system, multiple modulation types with different subcarriers should be studied. Besides, It is worth noting that the application of multi-modal and multi-task learning to DSMI has a good research prospect. For multi-modal DSMI, simultaneously using IQ sequence, constellation diagram, signal spectrum, and other types of data as the network entry to identify the signal can break the performance barrier caused by the thinness of single source data. For multi-task DSMI, The signal detection (the signal or no signal) can be considered as an auxiliary binary task processed in parallel with modulation identification, thereby expanding the model’s practicality and reducing the processing delay.

REFERENCES

1. Hsue, S.Z., Soliman, S.S.: Automatic modulation recognition of digitally modulated signals. In: IEEE Military Communications Conf, pp. 645–649.IEEE, Piscataway, NJ (1989)
2. Dobre, O.A., et al.: Survey of automatic modulation classification techniques: classical approaches and new trends. IET Communications 1(2), 137–156 (2007)
3. Long, C., Chugg, K., Polydoros, A.: Further results in likelihood classification of QAM signals. In: Proc. of IEEE MILCOM, pp. 57–61.IEEE, Piscataway, NJ (1994)
4. Dobre, O.A., Hammeed, F.: Likelihood-based algorithms for linear digital modulation classification in fading channels. In: 2006 Canadian Conf. on Electrical and Computer Engineering, pp. 1347–1350.IEEE, Piscataway, NJ (2006)
5. Hameed, F., Dobre, O.A., Popescu, D.C.: On the likelihood-based approach to modulation classification. IEEE Trans. Wireless Commun. 8(12), 5884–5892 (2009)
6. Shi, Q.H., Karasawa, Y.: Automatic modulation identification based on the probability density function of signal phase. IEEE Trans. Commun. 60(4), 1033–1044 (2012)
7. Swami, A., Sadler, B.M.: Hierarchical digital modulation classification using cumulants. IEEE Trans. Commun. 48(4), 416–429 (2000)
8. Aalam, N.W., Zhu, Z., Nandi, A.K.: Automatic digital modulation classification using genetic programming with k-nearest neighbor. In: Military Communications Conf, pp. 1731–1736.IEEE, Piscataway, NJ (2010)
9. Soliman, S.S., Hsue, S.Z.: Signal classification using statistical moments. IEEE Trans. Commun. 40(5), 908–916 (1992)
10. Park, C.-S. et al.: Automatic modulation recognition of digital signals using wavellet features and SVM. In: 2008 10th Int. Conf. on Advanced Communication Technology, pp. 387–390.IEEE, Piscataway, NJ (2008)
11. Peng, S.L., et al.: Modulation classification based on signal constellation diagrams and deep learning. IEEE Trans. Neural Networks Learn. Syst. 30(3), 1–9 (2019)
12. Peng, S.J. et al.: Modulation classification using convolutional neural network based deep learning model. In: 2017 26th Wireless and Optical Communication Conf, pp. 1–5.IEEE, Piscataway, NJ (2017)
13. Herrmann, A.P., et al.: CNN-based automatic modulation classification for beyond 5G communications. IEEE Commun. Lett. 24(5), 1038–1041 (2020)
14. Li, J.C., Qi, L., Lin, Y.: Research on modulation identification of digital signals based on deep learning. In: 2016 IEEE Int. Conf. on Electronic Information and Communication Technology, pp. 402–405.IEEE, Piscataway, NJ (2016)
15. Nie, J.B., et al.: Deep hierarchical network for automatic modulation classification. IEEE Access 7, 94604–94613 (2019)
16. Zhang, Z.F., et al.: Automatic modulation classification using convolutional neural network with features fusion of SPKVD and BJD. IEEE Trans. Signal Inf. Process. Networks 5(3), 469–478 (2019)
17. Hong, S., et al.: Deep learning-based signal modulation identification in OFDM systems. IEEE Access 7, 114631–114638 (2019)
18. Hong, S. et al.: Convolutional neural network aided signal modulation recognition in OFDM systems. In: 2020 IEEE 91st Vehicular Technology Conf, pp. 1–5.IEEE, Piscataway, NJ (2020)
19. Haring, L., Czyznik, A.: Automatic Modulation Classification Methods for Wireless OFDM Systems in TDD Mode. IEEE Trans. Commun. 58(9), 2480–2485 (2010)
20. Fadul, M.K.M., et al.: RF-DNA fingerprint classification of OFDM signals using a rayleigh fading channel model. In: 2019 IEEE Wireless Communications and Networking Conf, pp. 1–7.IEEE, Piscataway, NJ (2019)
21. Muquet, B., de Courville, M.: Blind and semi-blind channel identification methods using second order statistics for OFDM systems. In: Proc. 1999 IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Phoenix, pp. 2745–2748.IEEE, Piscataway, NJ (1999)
22. Muquet, B. et al.: A subspace based blind and semi-blind channel identification method for OFDM systems. In: 1999 2nd IEEE Workshop on Signal Processing Advances in Wireless Communications, pp. 170–173.IEEE, Piscataway, NJ (1999)
23. Muquet, B., de Courville, M., Duhamel, P.: Subspace-Based Blind and Semi-Blind Channel Estimation for OFDM Systems. IEEE Trans. Signal Process. 50(7), 1699–1711 (2002)
24. Yuan, H.M. et al.: Research on channel estimation for OFDM receiver based on IEEE 802.11a. In: 2008 6th IEEE Int. Conf. on Industrial Informatics, pp. 35–39. IEEE, Piscataway, NJ (2008)
25. Edfors, O. et al.: OFDM channel estimation by singular value decomposition. In: Proc. of Vehicular Technology Conf, pp. 923–927. IEEE, Piscataway, NJ (1996)
26. Zhao, M.H., et al.: Deep residual shrinkage networks for fault diagnosis. IEEE Trans. Ind. Inf. 16(7), 4681–4690 (2020)
27. Donoho, D. L.: De-noising by soft-thresholding. IEEE Trans. Inf. Theory 41(3), 613–627 (1995)
28. O’Hara, B., Petrick, A.: IEEE 802.11 Handbook: A Designers Companion. Standard Information Network, IEEE, Piscataway, NJ (2005)
29. He, K., Sun, J.: Convolutional neural networks at constrained time cost. In: 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 5353–5360. IEEE, Piscataway, NJ (2015)

How to cite this article: Zhang J, Hu S, Du Z, Wu W, Gao Y, Cao J. Deep learning-based digital signal modulation identification under different multipath channels. IET Commun. 2021;1–13. https://doi.org/10.1049/cmu2.12207